Calhoun: The NPS Institutional Archive

DSpace Repository

Theses and Dissertations 1. Thesis and Dissertation Collection, all items

1982

Network management of the SPLICE computer network.

Opel, Craig E.

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/20275

Downloaded from NPS Archive: Calhoun

DUDLEY KNOX LIBRARY

http://www.nps.edu/library

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

LIBRARY, NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA 93940

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

NETWORK MANAGEMENT OF THE SPLICE COMPUTER NETWORK

by

Craig E. Opel

December 1982

Thesis Advisor: N.F. Schneidewind

Approved for public release; distribution unlimited.

T208056

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Network Management of the SPLICE Computer Network		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis December 1982
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Craig Eric Opel		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE December 1982
		13. NUMBER OF PAGES 102
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Network management, Local area computer network, Monitoring technologies, Measurement tool, Network management reports, Performance analysis, Failure detection, diagnosis, and correction, Local computer network-Long haul network interface, Local computer network central monitoring site

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This thesis examines the network management functions required for a local computer network. Initially, general management considerations are addressed. These include: problem determination, performance analysis, problem management, change management, configuration management, and operations management. The sidestream, mainstream, centralized, decentralized, and hybrid network monitoring technologies are then discussed. An investigation of network measurement tools and their use in generating management reports is undertaken. The topics of analysis timing, performance measure utilization, and parameter

selection are considered. Procedures for detecting, diagnosing and correcting network component failures are presented. Solutions are proposed for problems associated with managing a local computer network-long haul network interface. Finally, a discussion of the mission, objectives, and responsibilities of a local computer network central monitoring site is undertaken.

Approved for public release; distribution unlimited.

Network Management of the SPLICE Computer Network

by

Craig E. Opel
Captain, United States Marine Corps
B.S., United States Naval Academy, 1976

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
December 1982

c.1

LIBRARY, NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA 93940

ABSTRACT

This thesis examines the network management functions required for a local computer network. Initially, general management considerations are addressed. These include: problem determination, performance analysis, problem management, change management, configuration management, and operations management. The sidestream, mainstream, centralized, decentralized, and hybrid network monitoring technologies are then discussed. An investigation of network measurement tools and their use in generating management reports is undertaken. The topics of analysis timing, performance measure utilization, and parameter selection are considered. Procedures for detecting, diagnosing and correcting network component failures are presented. Solutions are proposed for problems associated with managing a local computer network-long haul network interface. Finally, a discussion of the mission, objectives, and responsibilities of a local computer network central monitoring site is undertaken.

TABLE OF CONTENTS

I. INTRODUCTION 9
- A. ASSUMPTIONS 10
  - 1. Network Topology and Transmission Medium . 11
  - 2. Network Layer Protocol 11
  - 3. End to End Protocol 12
- B. LAN ARCHITECTURE 12
  - 1. Local Network Logical View 12
  - 2. Local Network Physical View 14
- C. NETWORK MANAGEMENT DISCIPLINES 14
  - 1. Problem Determination 16
  - 2. Performance Analysis 16
  - 3. Problem Management 16
  - 4. Change Management 17
  - 5. Configuration Management 17
  - 6. Operations Management 18

II. DESIGN ISSUES IN NETWORK MONITORING 19
- A. NETWORK MONITORING METHODOLOGIES 19
  - 1. Hardware Methodology 20
  - 2. Software Methodology 21
  - 3. Hybrid Methodology 23
- B. NETWORK MONITORING TECHNOLOGIES 23
  - 1. Sidestream Monitoring 25
  - 2. Mainstream Monitoring 26
  - 3. Centralized Monitoring 27
  - 4. Decentralized Monitoring 29
  - 5. Hybrid Monitoring 30
- C. CHAPTER SUMMARY 32

III. NETWORK MEASUREMENT TOOLS, AND MEASUREMENTS AND STATISTICS 33

A. NETWORK MEASUREMENT TOOLS 33
   1. Cumulative Statistics 34
   2. Trace Statistics 34
   3. Snapshot Statistics 35
   4. Artificial Traffic Generators 35
   5. Emulation 36
   6. Network Measurement/Control Center 37
B. MEASUREMENTS AND STATISTICS 37
   1. Host Communication Matrix 39
   2. Group Communication Matrix 40
   3. Packet Type Histogram 40
   4. Data Packet Size Histogram 41
   5. Throughput-Utilization Distribution . . . 41
   6. Packet Interarrival Time Histogram 42
   7. Channel Acquisition Delay Historgam . . . 42
   8. Communication Delay Histogram 43
   9. Collision Count Histogram 44
   10. Transmission Count Histogram 45
C. CHAPTER SUMMARY 45

IV. NETWORK PERFORMANCE ANALYSIS AND COMPONENT FAILURE 49
A. PERFORMANCE ANALYSIS TIMING 50
   1. Off-Line Analysis 50
   2. On-Line Analysis 51
   3. Instantaneous Analysis 52
B. LAN PERFORMANCE ANALYSIS 53
   1. Performance Measure Utilization 53
   2. Performance Parameter Selection 55
C. COMPONENT FAILURE 56
   1. Failure Detection 57
   2. Failure Diagnosis 61
   3. Failure Notification 63
D. CHAPTER SUMMARY 64

V. MANAGING THE LAN/DDN INTERFACE 68
  A. GATEWAY CONFIGURATION 69
  B. PACKET SIZING 72
  C. CONGESTION CONTROL 73
    1. LAN to LHN Packet Control 74
    2. LHN to LAN Packet Control 76
  D. ADDRESSING AND NAMING 77
  E. ACCESS CONTROL 79
  F. OTHER CONSIDERATIONS 80
  G. CHAPTER SUMMARY 82

VI. LAN CENTRAL MONITORING SITE 84
  A. MISSION OF A LAN MONITORING SITE 84
  B. MANNING AND ORGANIZATION OF A LAN CMS 86
  C. A NETWORK OPERATOR'S WORKBENCH 88
  D. OPERATORS ACTIONS: NORMAL CONDITIONS 89
    1. Initialization 90
    2. Utility Data Bases 90
    3. Operator's Displays 91
    4. Normal Management Activities 91
  E. OPERATORS ACTIONS: COMPONENT FAILURE 92
  F. CHAPTER SUMMARY 94

APPENDIX A: PROBLEM MANAGEMENT RECORD ENTRIES 95

APPENDIX B: CONFIGURATION MANAGEMENT RECORD ENTRIES . . 96

LIST OF REFERENCES 97

INITIAL DISTRIBUTION LIST 101

LIST OF FIGURES

1.1 Local Network Logical View 13
1.2 Local Network Physical View 15
2.1 Hardware Monitoring Device: Logical View 20
2.2 Hybrid Monitoring Device: Logical View 24
2.3 Centralized Monitoring 28
2.4 Decentralized Monitoring 29
2.5 Hybrid Monitoring 31

I. INTRODUCTION

One of the major objectives of any local network is to provide reliable communications facilities, reflected both in the continued availability of the network itself and in the lowest possible error rate as seen by individual processes [Ref. 14: p. 713]. To this we would add the requirements of high capacity and minimal end-to-end delay experienced by the user. We now submit what we feel is a responsible and complete definition of network management. Our definition includes: collection of measurements and subsequent statistics generation, hardware and software failure detection, diagnosis and correction, network performance analysis, and network parameter adjustment.

One school of thought advocates management of local area computer networks, while another feels that management, as we have defined it, is not required. We support the former of the two. The benefits to be gained from the management of a local computer network are numerous. We are able to reduce the impact of failures and increase network availability by detecting, diagnosing, and correcting hardware and software problems very quickly. Control and monitoring technologies allow network operators to anticipate problems. Rather than reacting, operators are able to analyze problems and take appropriate action to minimize them, or even preclude their occurence. Management of a local computer network gives us the ability to provide for capacity planning, manage the growth of the network, control costs, and eliminate redundant or unused capacity. We can also improve the networks performance and its availability to users by monitoring the network components and through evaluation of the network as a whole.

It is the author's intent to identify and discuss the tools required by network management for the attainment of these and other benefits. We will begin by describing a SPLICE local area computer network, followed by a discussion of six network management disciplines. Chapter 2 will present various network monitoring methodologies and technologies. In Chapter 3, we enter into a discussion of the measurement tools available to the operator and suggest ten management reports to be generated from collected data. Chapter 4 provides information on analysis timing, network performance measure utilization and parameter selection, and on component failure detection, diagnosis, and notification. Chapter 5 identifies and suggests solutions for the problems associated with managing the LAN/DDN interface. In Chapter 6, we conclude with a discussion of the mission, objectives, and responsibilities of a LAN central monitoring site.

A. ASSUMPTIONS

To productively discuss the topic of network management, it is important that a common base of understanding concerning the SPLICE (Stock Point Logistics Integrated Communications Environment) local area computer network be established. This section briefly describes the Network Layer Protocol proposed for the SPLICE LAN. This discussion will include; error detection, packet acknowledgement, collision detection, access control, bus control, retransmission technique, and packet format. Additionally, the network topology and physical transmission medium will be identified. Finally, a brief description of the proposed End to End Protocol will be discussed. A more detailed explanation of the SPLICE concept can be found in [Ref. 1].

1. Network Topology and Transmission Medium

The Ring, Star, Unstructured, and Global Bus topologies were discussed in [Ref. 2]. Primary considerations made during the selection of a topology were it's flexibility, reliability and simplicity. Understanding that the structure of the network must lend itself to change and reconfiguration, one author [Ref. 2: p.21] recommended that a global bus topology be adopted for the SPLICE local computer network.

Although a number of transmission mediums were discussed in [Ref. 2], no particular technology was recommended for all SPLICE network configurations. For this discussion of network management, it will be assumed that the transmission medium is coaxial cable and that a baseband technology is being utilized.

2. Network Layer Protocol

Decentralized control of the bus is the premis upon which all subsequent characteristics are based. Nodes access the network utilizing a random access contention mechanism with collision detection (CSMA/CD). Error detection is accomplished through the use of a cyclic redundancy checksum. The acknowledgement for a packet successfully received is undertaken by either sending a special acknowledgement packet or by including the acknowledgement with a data packet bound for the appropriate node. Upon detection of a collision, a node implements an adaptive binary exponential backoff retransmission technique. Finally there exists a single packet format for both data and control information, the specific type being identified in the packet type field [Ref. 2: p. 53].

3. End to End Protocol

TCP was utilized as a basis from which to develop the transport protocol. Justification for the use of TCP can be found in [Ref. 3]. A major consideration during the design of an end to end protocol was the assumption that SPLICE LAN'S would be connected to each other through the Defense Data Network. The fact that the end to end protocol currently planned for the DDN is TCP further excentuates the benefits to be derived by having a TCP based transport protocol. Investigation shows that if TCP is used in the strictest sense without any modification as the local transport control protocol, simple internetwork communication will be achieved at the expense of suboptimal intranetwork performance [Ref. 2: p. 78].

B. LAN ARCHITECTURE

This section depicts and briefly describes the logical and physical views of the SPLICE LAN. These diagrams are included in order to provide a visual representation which may be referred to during the discussion of network management throughout the thesis.

1. Local Network Logical View

The six boxes along the top of figure 1.1 are identified as operation functions implemented in software modules. The three boxes to the right and in the middle represent support functions implemented in software modules.

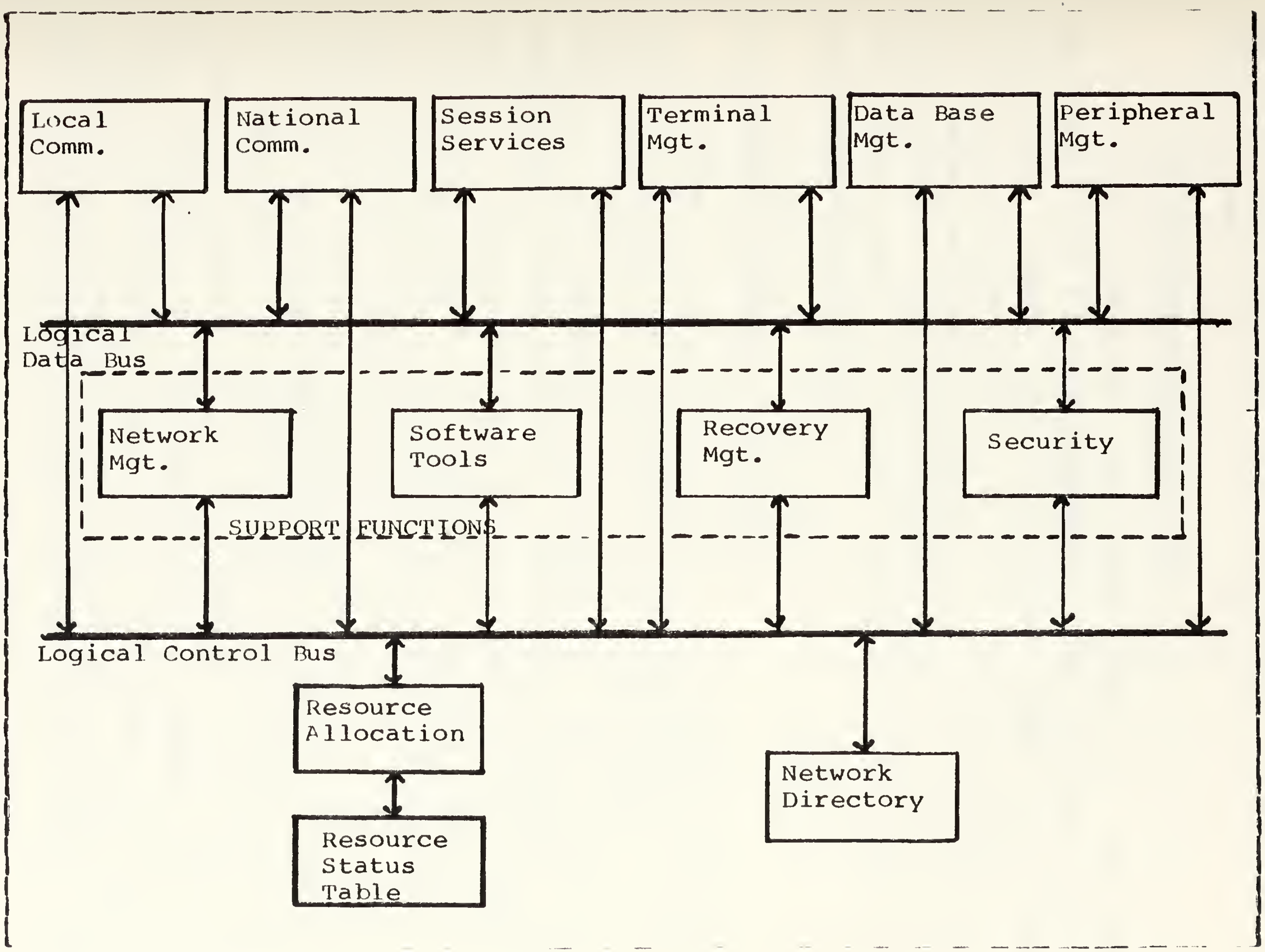

Figure 1.1 Local Network Logical View.

Control messages flow along the logical control bus while data messages flow along the logical data bus. A more detailed explanation of these functional modules can be found in [Ref. 4].

2. Local Network Physical View

There exists only one physical bus upon which will flow both control and data messages. The functions identified in the logical view of the network have been assigned to specific minicomputers. As can be seen in figure 1.2, the network management function has not been identified. Theoretically, this function could reside in one or all of the network nodes. An indepth discussion of this topic will be undertaken in Chapter 2.

C. NETWORK MANAGEMENT DISCIPLINES

If viewed as a single module, the network management function appears quite complex. Different aspects of the function appear to overlap, while others appear to be disjoint and unrelated. In order to more effectively analyze the various aspects of the network management function, a disaggreagation of the function into unique, identifiable modules is undertaken. Freeman proposes six distinct management disciplines associated with managing the components of a computer network [Ref. 5: p. 91]. These disciplines include; problem determination, performance analysis, problem management, change management, configuration management, and operations management. The purpose for presenting these disciplines is twofold; First, to create more managable and understable modules through which the concept of network management can be discussed, and second, to provide a foundation upon which various network management techniques can be analyzed throughout the thesis. Each

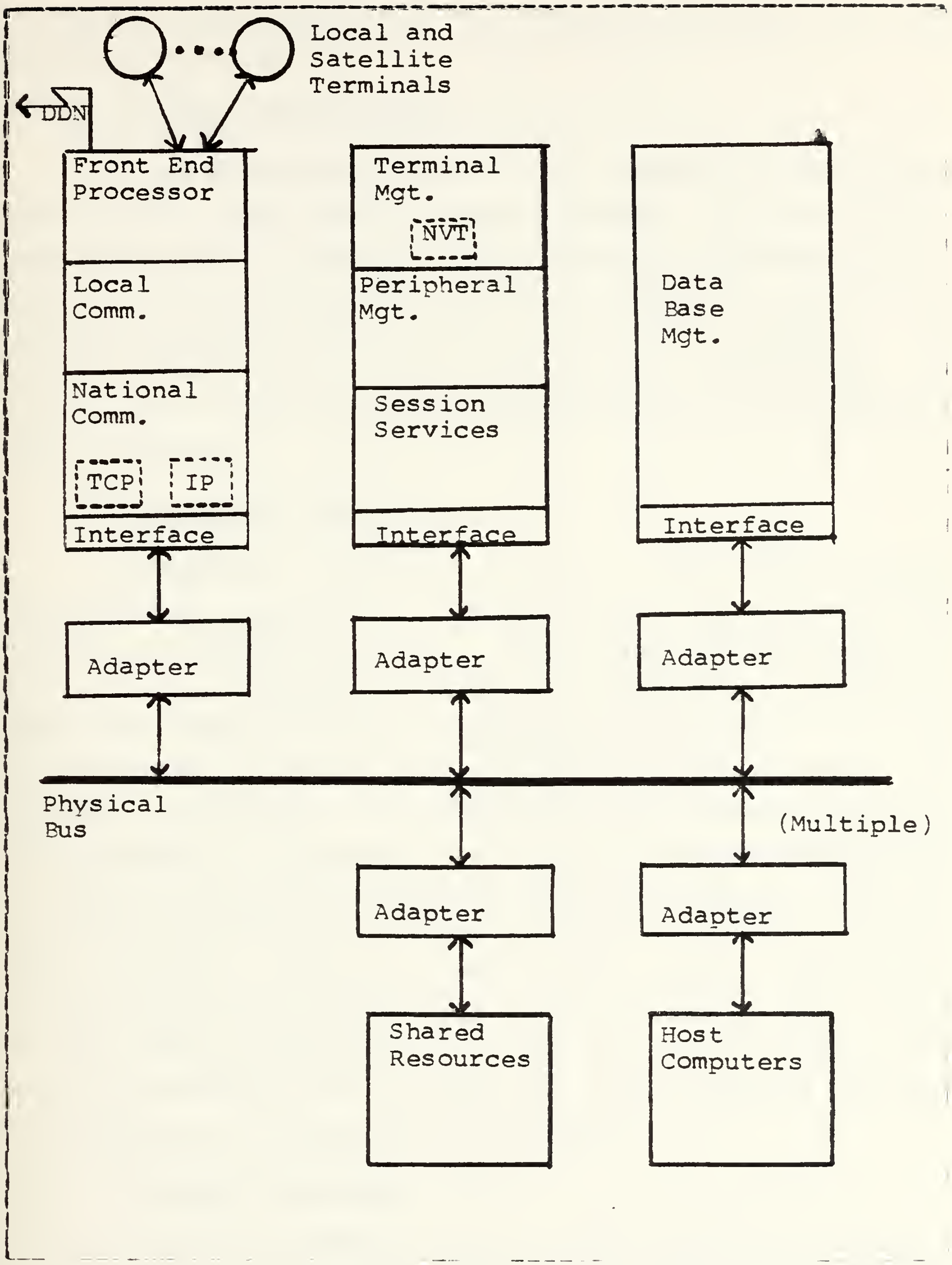

Figure 1.2 Local Network Physical View.

of these disciplines will be briefly described in the following sections.

1. Problem Determination

Problem determination is the process of identifying a failing or down component of the network so that corrective action may be taken. It includes; awareness that a problem exists, isolation of the problem to a particular element, identification of what caused the problem, and determination of the correct organization, individual, or vendor who is responsible for the correction of that specfic type of problem.

2. Performance Analysis

Peformance Analysis deals with quantifiably answering the question of, 'How well is the network doing what it is supposed to do?'. It provides for the measurement of certain dependent variables throughout the network. These measurements are then compared to criteria that have been previously established by some other means (e.g. by mathematical models). By observing the variance between these figures, a snapshot of the network's performance can be obtained for that particular instant in time. A number of variables measured can be classified as "tuning" statistics. Certain parameters exist which can be adjusted by network operations personnel in order to effect the values of these tuning statistics. In this way, we can affect both network performance and the quality of the service provided by the network as perceived by the user.

3. Problem Management

Problem Management concerns the reporting, tracking, and resolution of problems that affect a user's or process' capability to communicate with any other user or process.

Establishment and maintenance of a problem database can be accomplished in a number of ways. Problems may be documented manually utilizing pencil and paper. They may be recorded semi-automatically through manual entry into a database. Or, problems may be recorded automatically through the interaction of the problem management module with the problem determination and performance analysis modules. The method chosen through which network problems will be recorded should provide for data consistency, real time information or nearly so, user accessibility, and minimal operations personnel envolvement. A list of possible entries for inclusion in a problem record is provided in Appendix A.

4. Change Management

Changes made to a network component that are not promulgated throughout, or made available to the network may lead to substantial delay when communicating with that element or even make that element inaccessible. Change management precludes these events from occuring by reporting, tracking, obtaining approval for, and verifying the implementation of changes in network components [Ref. 5: p. 91]. Pencil and paper, or manual entry into a database are two methods by which change management may be accomplished.

5. Configuration Management

Configuration management provides for the creation of a database which contains the past, present , and future physical and logical characteristics of all network elements [Ref. 5: p. 91]. Included in this would be the SPLICE mini-computers, host computers, shared resources, the subnetwork, and pertinent information concerning any connected networks. The configuration management database should be accessible

by both software modules and users as needed. Updating and maintenance of this database could be accomplished in the same manner as the problem management database. It is this researcher's opinion that configuration management could most efficiently be accomplished utilizing automated techniques which are based on the interaction of the various network management modules. A list of entries that may be included in a configuration management record of a network component is included in Appendix B.

6. Operations Management

Operations management supports the remote manipulation of various network elements [Ref. 5: p. 91]. Some of the forms this manipulation takes includes; testing a piece of hardware such as an adapter, testing specific software such as a process which counts the number of times an individual packet attempts to access the channel before it is successful, adjusting parameters in order to effect the values of certain dependent variables which characterize the performance of the network, and starting up a remote process within a node which acts as an artificial traffic generator. Additionally, during the process of network reconfiguration, this management function supports the remote loading of software into the appropriate network element.

II. DESIGN ISSUES IN NETWORK MONITORING

Measurements allow us to gain valuable insight regarding network usage and behavior [Ref. 6: p. 1439]. They provide a means to evaluate the performance of the implemented protocols. Additionally, they give the designer the ability to detect network inefficiencies and identify design flaws. On an operational level, measurement provides the statistics upon which the network is tuned through adjustment of appropriate parameters. In a global sense, measurement can be seen as the foundation upon which network management is based. Hamming expresses the importance of measurement in the statement, "It is difficult to have a science without measurement" This emphasis on an accurate measurement capability assists in understanding why such elaborate and complex measurement techniques have been devised for experimental and operational networks.

Before any type of measurement is conducted of a network or it's associated components, two basic questions must be answered. They are, 'What is to be measured?', and 'Why should the measurement be taken?'. These questions will be addressed in Chapters 3 and 4 respectively. At this time, an explanation of basic monitoring methodologies will be undertaken, followed by a discussion of current monitoring technologies.

A. NETWORK MONITORING METHODOLOGIES

Currently, there exists three basic methodologies utilized as the foundations for the creation of various network monitoring technologies. These three methods are hardware monitoring, software monitoring, and hybrid

monitoring. These methods will be discussed for the purpose of establishing a basis upon which the monitoring technologies may be analyzed.

1. Hardware Methodology

A pure hardware monitor is a unit that is both physically and logically distinct from the network component being measured [Ref. 7: p. 57]. The interface between the monitor and the component is a physical probe used for the collection and passing of electronic signals from the component to the monitoring device. Figure 2.1 depicts a generalized hardware monitoring device [Ref. 7: p. 57].

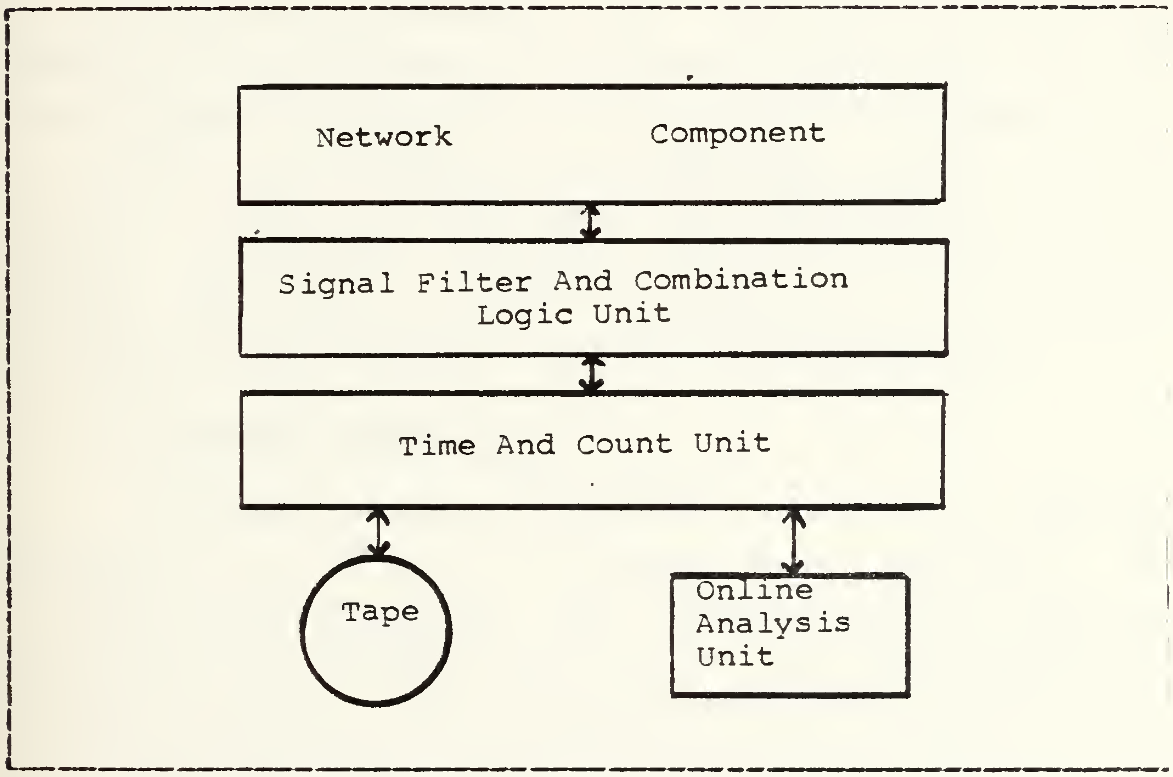

Figure 2.1 Hardware Monitoring Device: Logical View.

The critical item needed for a hardware monitor is an electronic signal that indicates the occurence of an event [Ref. 7: p. 57]. Since many signals to be monitored are of a relatively low voltage, one must consider that the introduction of a monitoring device may disturb the normal operation of the circuit being monitored. To preclude this, a high impedance probe can be utiilzed. The signal observed by the probe is amplified and passed to a signal filter and combination logic unit. The task of the signal filter and combination logic unit is to mask and combine signals received from various probes. This output is then sent to a time and count unit. Here, the duration of a specific signal, or the number of times a certain signal occures can be recorded. Having collected the required data appropriate for the test being conducted, the contents of the time and count unit can be directed to a mass storage device for off line analysis or, directly to a user for on-line analysis.

The main advantage of a hardware monitor is it's ability to sense a wide range of hardware and software events. In addition to cost, the main disadvantage of a hardware monitoring device is it's limited ability to detect the stimulus for the set of signals it is monitoring.

2. Software Methodology

Although various definitions exist, a software monitor can be viewed as a process which resides in the component being monitored. Two types of software monitors exist which are appropriate for the task of monitoring a computer network. They are the interrupt-intercept methodology and the sampling methodology [Ref. 7: p. 56].

The interrupt-intercept methodology embraces the idea of carrying out some type of monitoring activity every time the state of the particular resorce in which the monitor is resident changes. The monitoring routine is

invoked whenever an interrupt is generated. The scheme calls for intercepting each interrupt as it occures, directing it to a monitoring routine where the interrupt is analyzed and appropriate monitoring functions activated, and finally, passing the interrupt to it's intended destination. This monitoring methodology has the distinct advantage of allowing measurements to be taken as an integral part of the system rather than as a lower level application program. Substantial amounts of processing time and memory utlization are required for this method. Additionally, it also requires that the software monitoring program run at a very high priority to prevent other interrupts from deactivating the monitor [Ref. 7: p. 57].

The sampling methodology treats the software monitoring program as a normal user program for a multiprogramming system. The activation of the monitoring program may be accomplished by the component resident operating system, by another monitoring application program, or by network operations personnel. This activation may occure at random intervals, scheduled intervals, or a combination thereof. The selection of inter-sample periods is critical in that it must not be synchronized with the occurence of events which are being measured by the monitor [Ref. 7: p. 57]. As with the interrupt-intercept methodology, a significant amount of processor time and memory space may be required.

The principal advantage of the software monitors presented above is their ability to associate occurances of measured events with their causes. The primary disadvantage is their requirement for substantial resource utilization. The strengths of the hardware and software monitoring methodologies have been combined and their weaknesses eliminated through the use of a hybrid approach.

3. Hybrid Methodology

In contrast to the hardware monitoring methodology, the hybrid approach to monitoring does not view the hardware monitoring device as being invisible to the network component. The hybrid methodology utilizes a microcomputer to control the functions of the hardware monitoring device in response to data gathered by hardware probes. Figure 2.2 represents the logical view of a hybrid monitoring device. The data channel provides a means by which the software monitor resident in the device being monitored can communicate with the hardware monitoring device. Along this channel can pass interrupts and messages concerning the occurence of software events within the component. These can then be associated with signals sensed by the probes of the hareware monitoring device. This overcomes the strict hardware monitoring methodology's inability to associate a signal with a specific event occurance within the network component. Additionally, the problem of component resorce utilization associated with the strict software monitoring methodology is overcome by the transition of various monitoring functions from the network component to the hardware monitoring device.

Technologies for the location of monitoring capabilities within a computer network implicitly utilize one of the methodologies, or a variation thereof, discussed above. A number of these technologies will be discussed in the following section.

B. NETWORK MONITORING TECHNOLOGIES

There are certain considerations that should be addressed when selecting a monitoring technology. Initially, a decision has be be made on whether of not a record of every occurance of a certain event should be made,

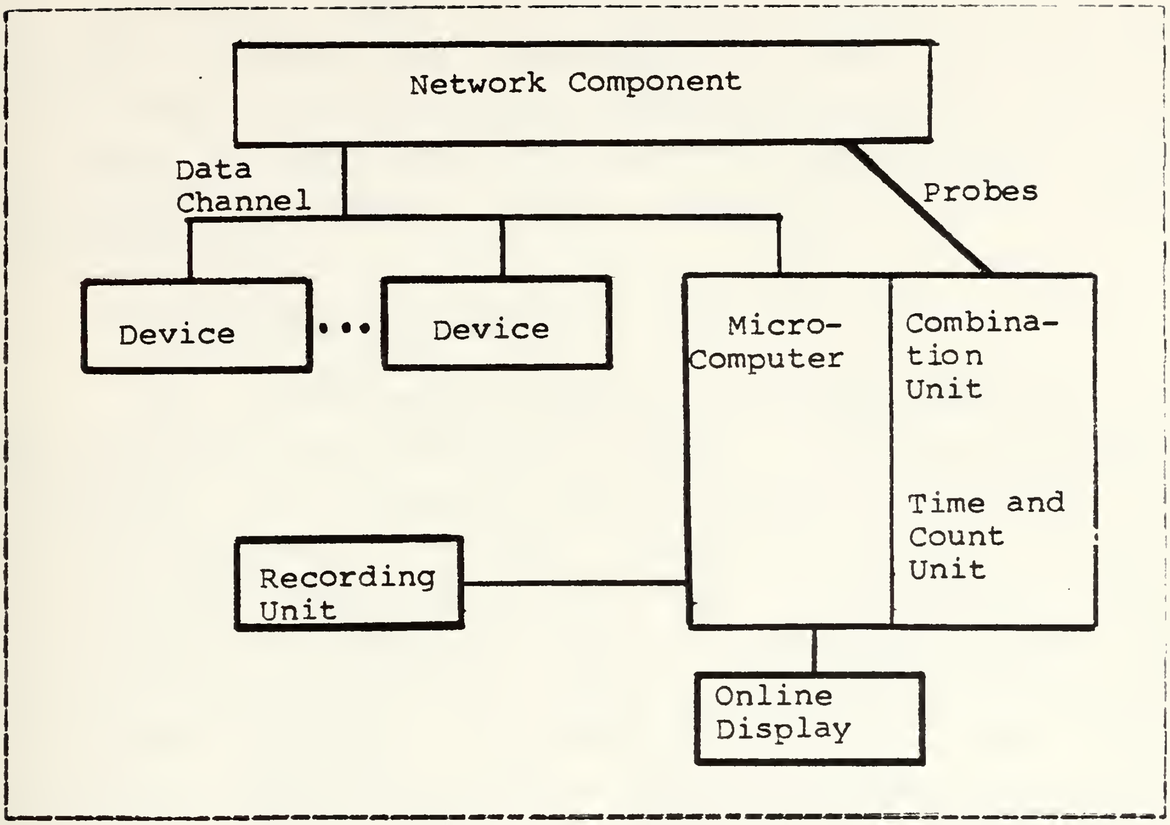

Figure 2.2 Hybrid Monitoring Device: Logical View.

sometimes called trace monitoring [Ref. 7: p. 53], or to collect samples from the network at selected intervals of time. Timing considerations for the NBSNET measurement system indicate complete measurement is possible [Ref. 8: p. 726]. It's architecture, being similar to that of the SPLICE LAN would seem to indicate that complete measurement would be possible for the SPLICE LAN. Whether this would be desirable or practical are questions that remain to be addressed. The technique selected must be able to monitor both hardware and software components individually and any combination thereof. The level of monitoring to be conducted must be determined. Does the technology under consideration provide the capability of both a macroscopic

and microscopic level of monitoring? Is the monitoring technique capable of supporting a real-time analysis requirement? To what degree does the monitoring technique introduce artifact into the system? Other items to be considered include; clock resolution and clock synchronization.

1. Sidestream Monitoring

The sidestream monitoring technology [Ref. 5: p. 92], requires that probes be attached to the side of network components. By attaching these probes to the 'side' of network components, we mean physically placing them such that they may sample and analyze data from physical interfaces within the component, and at the interface between the component and network bus. These probes extract and analyze data from physical interfaces established with these elements. Additionally, the sidestream technique obtains information about the network interface and the subnetwork through the use of a meaurement module resident in the adaptor. Information gathered by these probes and modules may be sent to a network monitoring center, or to a set of management programs via a secondary channel which is frequency-division multiplexed onto the same circuit being used by the primary data channel.

A major advantage of the sidestream technique is it's ability to alert network operations personnel of certain types of problems without interferring with normal data traffic. Certain tests may also be undertaken which utilize this secondary channel. In this way, isolation testing may take place without disrupting the primary data channel. Even though a secondary channel assists in isolating and correcting certain problems, there still remain certain tests that must utilize the primary data channel for their accomplishment. This has been found to be

one of the major problems of the sidestream technique due to the fact that, during the conduct of these tests, the network is unavailable.

The sidestream monitoring technology presented here is a subset of a more encompassing network management philosophy. Our discussion has briefly touched on the topic of component failure identification. This was determined to be necessary in order to more clearly define and explain the advantages and disadvantages of this technology. This subject will be addressed again when a discussion of various techniques for identifying, isolating, and correcting failing network components is undertaken in Chapter 4

2. Mainstream Monitoring

The mainstream monitoring technique operates thru the use of hardware and software implemented among existing network components. These additions provide data to a network monitoring center or a set of network management programs. Notification of problems existing within the network is accomplished through the generation of asynchronous problem messages. These messages are communicated as normal data traffic on the primary data channel. Data provided by these asynchronous problem messages is usually sufficient to isolate a problem to a particular component without further problem isolation tests such as those required by the sidestream method. Error records within a problem message contain specific information concerning the problem being reported. Information contained in the error records is generated by testing modules resident in the network components, which are invoked upon problem recognition. If information contained within the problem record is insufficient to isolate the cause of a specific problem, additional isolation testing is initiated.

The major advantage of the mainstream monitoring technique is it's ability to isolate and diagnose a problem based upon information contained in the problem message. A problem with this technique evolves around the requirement for these problem messages to utilize the primary data channel for transmission to the central monitoring site. If an adaptor is down through which the message must pass, or if the subnetwork congested, the problem message may experience some delay before being communicated to the central monitoring site.

Like the sidestream technique, the mainstream technology presented here is a subset of a more encompassing network management philosophy. Discussion of problem identification and isolation was included for clarification purposes. Additional discussion on the subject of component failure identification, isolation, and correction will be undertaken in Chapter 4.

3. Centralized Monitoring

A broadcast network lends itself naturally to a centralized measurement approach [Ref. 8: p. 725]. Centralized monitoring requires modification of the adaptor connecting the processor which houses the network management function to the bus. Through this modification, the adaptor can monitor all packets on the network. Some of the information which can be extracted and determined from monitoring packets transiting the network includes; packet size, number of packets of each type transmitted, and interarrival time since last packet. Since the modified adaptor simply makes a copy of the passing packet, extracts the required information from it, and discards the copy, no artifact is being introduced into the system.

Certain important information cannot be obtained utilizing the centralized monitoring technique. The time between arrival of a packet at the network interface and it's subsequent transmission onto the network is only available at the interface. Thus we have no measurement of the effectiveness of our access protocol. Although a collision on the network can be detected by the central monitor, it is not capable of determining which nodes packets were involved in the collision.
the central monitor is biased. This is caused by the propagation delay between the sending adaptor and the monitoring adaptor. Figure 2.3 depicts a local network with centralized monitoring.

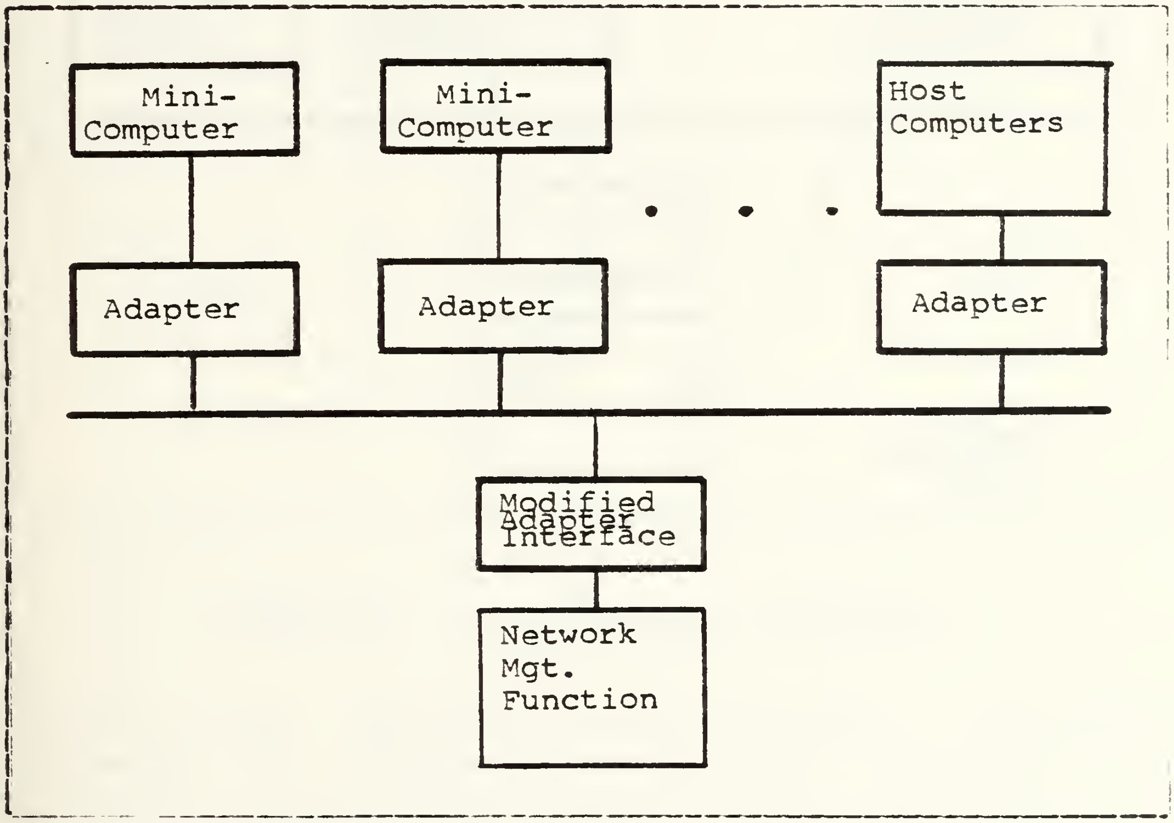

Figure 2.3 Centralized Monitoring.

4. Decentralized Monitoring

Figure 2.4 represents a decentralized monitoring scheme. In using this approach, the burden of network monitoring is placed on each individual interface. The functions of the central monitoring site no longer include monitoring. The tasks performed by the central monitoring site are now restricted to data collection from the adaptors, data reduction and data analysis. Measurement information is obtained by the central monitoring site through the receipt of information packets generated by the individual adaptors. Transmission of measurement information may be as frequent as with every packet. Other

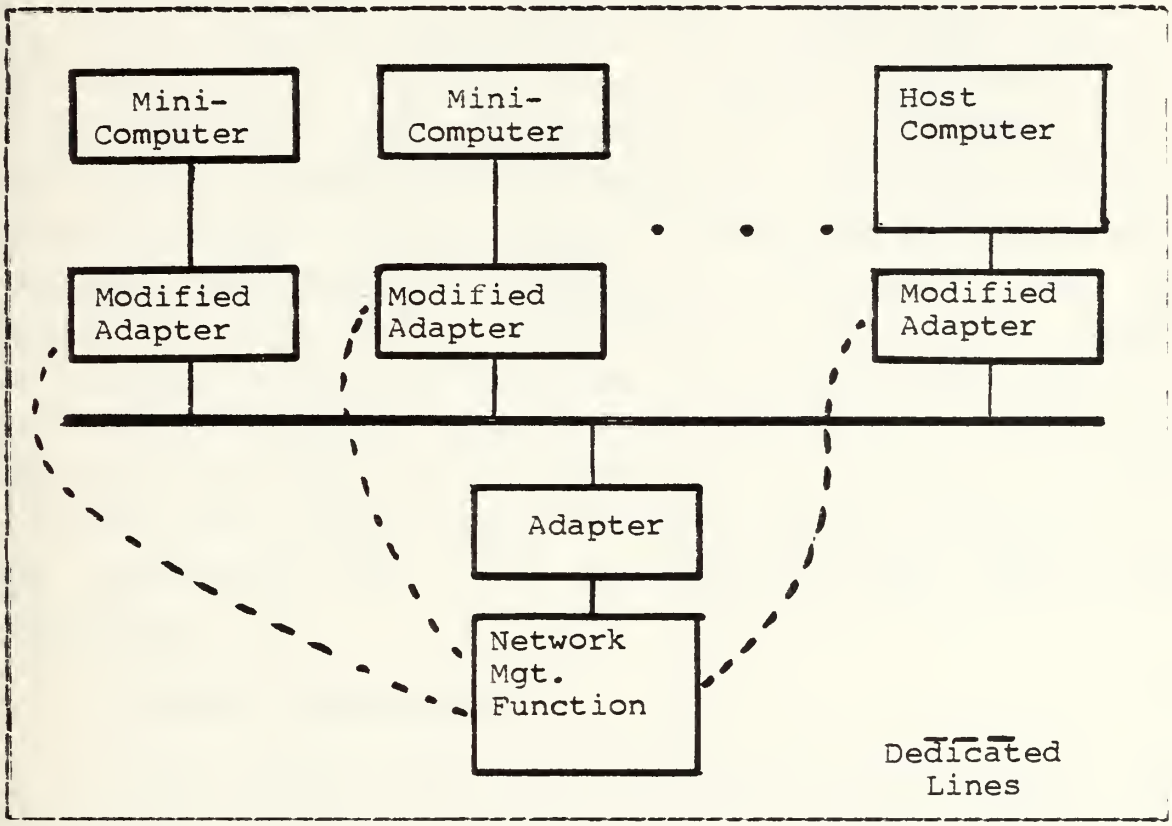

Figure 2.4 Decentralized Monitoring.

protocols call for the transmission of measured information after a certain amount of time has elapsed, or, after a certain number of events have occured.

With a decentralized approach all information about the network traffic is available [Ref. 8: p. 725]. Information about collision induced delays and collision counts can be obtained from each adaptor. Exact times for packet transmission and receipt are available. Another positive attribute of decentralized monitoring is the ability to identify those nodes whose packets were involved in a collision. To provide this enhanced service, additional memory and real time clocks must be incorporated into each network interface. Additionally, the periodic transmission of data to the central monitoring site requires overhead communication. If sent over dedicated lines, as depicted in Figure 2.4 , extra costs are incurred. If these information packets are sent over the primary data channel, artifact is introduced into the system. Finally, since this technique requires that all adaptors in the network possess a greater than normal degree of intelligence, implementation and maintenance tend to be more costly than centralized monitoring.

5. Hybrid Monitoring

The hybrid monitoring technique grew out of the advantages and disadvantages of the centralized and decentralized technologies. In this approach, as much information as possible is collected by the central monitoring site. Only those measurements unobtainable by the central monitor are measured by each network interface. This allows for minimal modification to the network interface. Figure 2.5 represents the hybrid monitoring technique.

The transmission of data to the central monitoring site is initiated upon the termination of a logical connection. Implementation of this protocol reduces the introduction of artifact into the system.

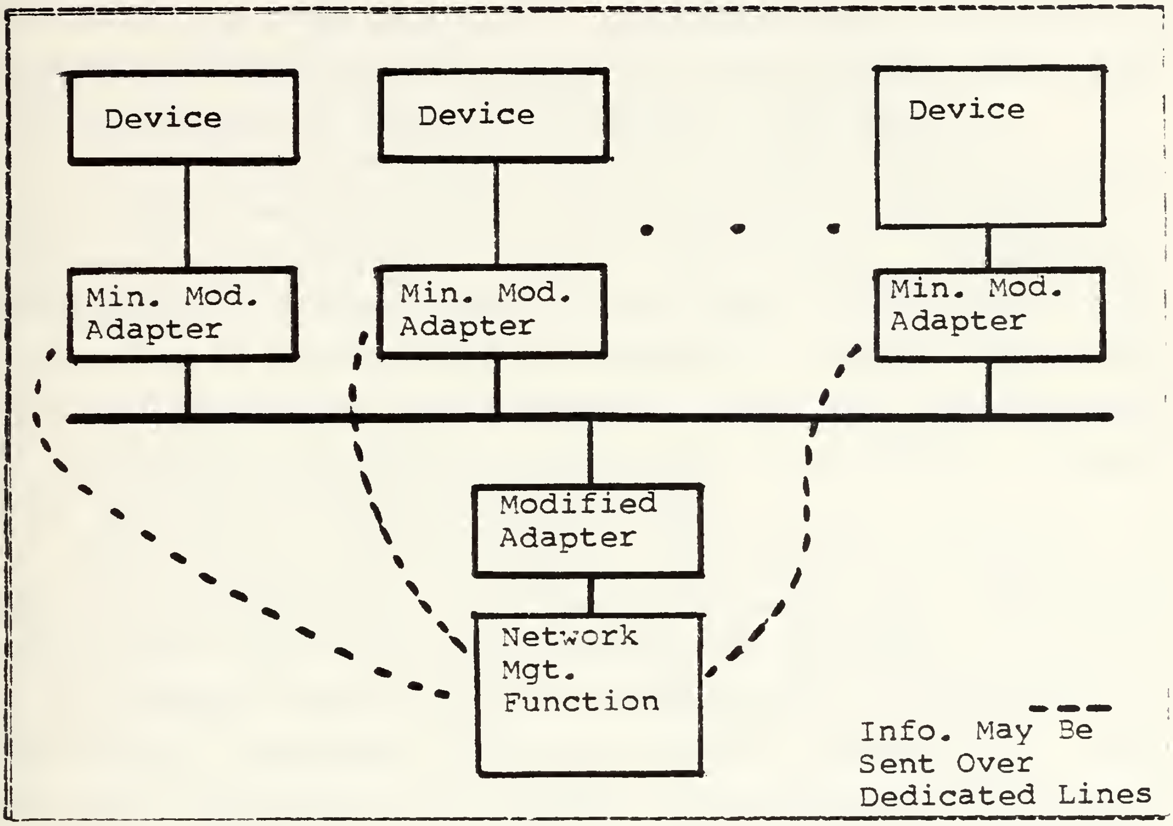

Figure 2.5 Hybrid Monitoring.

In combining the advantages and eliminating the disadvantages of centralized and decentralized monitoring, the hybrid monitoring technology has provided the network with an accurate and comprehensive measurement and monitoring capability. One disadvantage deals with the complexity of coordinating the analysis of decentralized and centralized measurement [Ref. 8: p. 725].

C. CHAPTER SUMMARY

This Chapter began with an explanation of hardware, software, and hybrid monitoring methodologies. Strengths and weaknesses of each were discussed. Those attributes, both positive and negative, associated with software and hardware monitoring were found to be the criteria upon which the development of the hybrid approach was based.

In the second section of this Chapter, implementations of the basic methodologies were presented. The monitoring technologies addressed were; sidestream, mainstream, centralized, decentralized, and hybrid monitoring. The discussion of each technology included; a brief explanation of the operation of the monitoring technique, presentation of advantages and disadvantages, and in some cases, comparison to other monitoring technologies.

Each one of the monitoring technologies presented is capable of providing adequate monitoring and measurement capabilities for use by the SPLICE LAN management function. It is proposed that the hybrid technique be adopted as the monitoring technology utilized by the SPLICE LAN. This technique emphasizes the concept of minimizing data collectionat network interfaces. Only those measurements unobtainable by the central monitor would be gathered by the adaptors. As in the mainstream monitoring technology, each adaptor would be capable of problem detection and invoking local test modules which would gather data concerning the problem for subsequent transmission to the central monitoring site. Data collected by the adaptors would be sent to the central monitoring site as administrative packets over the primary data channel. In addition to the transmission of routine measurement information upon the termination of each logical connection, problem messages, similar to that implemented by the mainstream technology, will be transmitted asynchronously to the central monitoring site.

III. NETWORK MEASUREMENT TOOLS, AND MEASUREMENTS AND STATISTICS

To this point, the general architecture of a SPLICE LAN, along with a proposed monitoring methodology has been presented. Now that a method exists which allows us to obtain information from the network, the focus of this thesis changes to address the question; What measurements and statistics should we be able to derive from the network in support of experimental and operational functioning? An attempt will not be made to itemize measurements and statistics required for the accomplishment of each specific experimental or operational endevor. Rather, a discussion of basic measurement tools will be undertaken, followed by the identification and explanation of measures and statistics appropriate for use in managing local area networks which must interface with the DDN and where control of the dominent DDN does not come under the authority of the LAN managers.

A. NETWORK MEASUREMENT TOOLS

In order to evaluate the performance of a network, and to identify down or failing components, several measurement tools must be available. These tools are: cumulative statistics, trace statistics, snapshot statistics, artificial traffic generators, emulation, a network measurement center which includes control, collection and analysis of data, and a network control center which accomplishes status reporting, monitoring, and controling the network. These latter two tools may be combined into a single entity which is sometimes called a monitoring center. Each of these tools will be addressed in the following section.

1. Cumulative Statistics

Cumulative Statistics consist of data regarding a variety of events, accumulated over a given period of time. These are provided in the form of sums, frequencies, and histograms [Ref. 6: p. 1439]. This tool is one that should be included in the capabilities of the SPLICE local area computer network measurement facility. Since some cumulative statistics can become quite long, it is wise to control their transmission to the central site in some way. One approach might be to designate certain items within a cumulative statistical message as being optional. This provides network operations personnel with many measurement capabilities, yet precludes the formulation and transmission of excessively long cumulative statistical messages.

2. Trace Statistics

Trace statistics allow network operations personnel to literally follow a packet through the network and to learn of the route that it takes and the delays it encounters [Ref. 10: p. 633]. Obviously, in a bus oriented network, there does not exist a requirement to identify the route a packet has taken to it's destination. Although more applicable to a packet switched store and forward network, certain aspects of a trace mechanism may prove useful in a local area network. Such an area might include possibly timestamping the packet as it arrived at the adaptor from a processor, and subsequently recording the time the packet was successfully transmitted. Additionally, the packet could be timestamped when it arrived at the destination adaptor and subequently record the time at which the packet is forwarded to the resident processor. These statistics can then be forwarded to a central monitoring site upon demand or at some predetermined time.

3. Snapshot Statistics

Snapshot Statistics provide an instantaneous look at a device showing it's state with regard to various queue lengths and buffer allocation [Ref. 6: p. 1440]. In a high speed, dynamic environment such as a local area network, these types of statistics can prove valuable in the evaluation of certain protocols. Evaluation of a network access protocol could be conducted by observing the length of the 'packets ready for transmission queue'. Additional information that could be contained in a snapshot of a particular network component or set of components are processor queue lengths, storage allocation, and status of adaptor buffers for receipt and transmission of packets.

4. Artificial Traffic Generators

The use of artificial traffic generators provides network operators with the ability to create streams of packets with specified durations, inter-packet gaps, packet lengths and other appropriate characteristics [Ref. 6: p. 1440]. This tool plays a major role during the implementation of the LAN. In the absence of sufficient traffic to test certain aspects of the network, artificial traffic generators provide the mechanism through which varying network load conditions can be simulated. This provides a more realistic environment in which testing may be conducted, and provides a mechanism that can be used to identify network problem areas. By doing this, we are able to effect modifications to the LAN while it is in it's infancy rather than attempting to make changes when production activities are heavy and corrections more expensive. Additionally, artificial traffic generators may be used to test and analyze various network protocols. This is accomplished through the generation of identical transmission

strings, thereby providing a basis upon which the performance of the various protocols can be compared.

Amer [Ref. 8: p. 726], has identified five capabilities that should be possessed by an artificial traffic generator. These include the ability to: 1) generate packets with a constant, uniform, or Poisson size distribution, 2) generate packets with constant, uniform, or exponential interarrival times, 3) direct packets to any specified destination, 4) communicate with the monitoring system to synchronize traffic generation and data collection and 5) permit on-line operations personnel control.

5. Emulation

Emulation is the creation of an illusion that there exists more components of a certain kind in the network then actually exists. Each one of these "fake" components is capable of displaying the characteristics of, and performing the functions of a "real" physical component of that type. Emulation is required when there are not enough network components to provide sufficient traffic generation and a range of nodal characteristics. In supplimenting these areas, emulation gives the operator a better understanding of network behavior under various configurations . Closely related to this is the use of emulation in conjunction with capacity planning. Through emulation, we are able to determine what effect a change to the network configuration will have on various performance measures. A situation in which emulation might be employed would be to determine the affect of adding or deleting a host processor from the SPLICE local area computer network.

6. Network Measurement/Control Center

In the early, experimental days of the ARPA Computer Network, there existed physically separate measurement and control centers. This allowed for continual experimentation with the network from the measurement center, while actual control of the network was conducted from the control center. These two functions of measuring and controlling have been combined, and will be undertaken by a single monitoring center for the Defense Data Network [Ref. 11: p. 95-102]. The responsibilities of a Network Measurement/Control Center include: controlling the measurement facilities, collecting and analyzing data, generating status reports, and monitoring and controlling the network. These responsibilities, as they apply to a local computer network, will be addressed in much greater detail in Chapter 5.

B. MEASUREMENTS AND STATISTICS

Now that the measurement tools have been identified and discussed, the question arises, 'How do we select and implement the appropriate tools for the measurement task at hand?'. Before this subject can be addressed, the answers to two questions posed in Chapter 2 must be determined. Those questions were: Why should the measurement be taken? (i.e. What managerial and research questions are to be answered by the measurement?), and, What is to be measured? (i.e. What specific network characteristics must be measured in order to satisfy these questions?).

An approach to answering these two questins is as follows. Initially, it is appropriate that the object of the measurement operation be defined. This entails the identification of some specific area of the network to be investigated. In conjunction with this, the goals of the

measurement operation are solidified. The next step is to select those performance measures that best characterize the area of the network being studied. Finally, the specific measurement tools most appropriately suited for the measurement operation are identified and selected for implementation.

Goals of measurement operations are usually motivated by a desire for software verification, for performance evaluation and verfication, to obtain feedback for system design iterations, to identify down or failing components, and to study user behavior and characteristics. Performance measures can be catagorized as basic, special, or composit. Examples of basic measurements include throughput and delay. When examination of a specific procedure is required, specialized measures must be used to compliment the basic measures. They are aimed at measuring a specific attribute of a specific network component. Finally, in order to analyze some global system properties which cannot be easily described by throughput and delay, it becomes necessary to aggregate a set of measurements that have been taken over a specific monitoring period. This aggregation of measures is called composite measurement. Examples of composite measures include, fairness, congestion protection, stability, robustness of network algorithms to line errors, and reliability of a network configuration with respect to component failures [Ref. 6: p. 1443].

Returning to the subject of measurement tool selection and implementation, we find that a subset of these tools have been utilized in obtaining spe cfic data from an operational local area computer network [Ref. 8]. In describing certain reports generated from data collected throughout this network, the researcher will be attempting to show how the tools are selected and integrated inorder to provide network operations personnel with accurate, timely and

sufficient information upon which the management of the network may be based.

It would be infeasible to identify and provide rationale for every measurement that could be taken from a local area network. Additionally, this list would be a dynamic one, dependent upon the goals and objectives of the specific network analysis operation to be undertaken. For these reasons, an established measurement capability for a local area computer network will be investigated in an attempt to bring forth and discuss the implementation of various measurement tools as they pertain to the SPLICE LAN. Ten performance reports have been implemented for measuring NBSNET traffic [Ref. 8]. Each report is classified as either traffic charaterization or performance analysis type. Traffic characterization reports indicate the workload placed on the system. Performance characterization reports indicate the time delays, utilizations, etc., which result from a given load and network configuration. They describe the dependent variables that are observed rather than controlled, and are used for tuning the network and making performance comparisons [Ref. 8: p. 726]. At this time, a brief description of each report will be given, along with appropriate comments relating to requirements of, and recommentations for the SPLICE LAN.

1. Host Communication Matrix

The Host Communication Matrix indicates the traffic flow between connected nodes. For each node, data tabulated includes: the total number of packets, data packets and data bytes received from and sent to all other nodes. From this, the proportion of data packets to total packets, and data bytes to total bytes are determined. Utilizing the monitoring technique proposed for the SPLICE LAN in Chapter 2, this information could be obtained by the central monitoring

center through it's tap into the channel. In this way the number of bytes in a packet can be counted by the monitoring center, and the source, destination, and packet type determined from the header. Additionally, a summary of the total network traffic is made available which includes total packets, data packets, and data bytes transmitted, and the mean number of data bytes per packet.

2. Group Communication Matrix

The Group Communication Matrix indicates the traffic flow between any user defined groupings of nodes. The same type of information tabulated by the Host Communication Matrix is recorded by the group communication matrix. The possible extension of this concept to include the recording of the traffic flow between various user designated processes may prove more valuable than the information originally seen as the product of this report. An example of this would be the recognition that a number of processes utilize the same data file on a regular and possibly concurrent basis. Assuming this data is currently kept on a tape storage device (which is not out of the question in a government installation), and possessing the information provided by the Group/Process Communication Matrix, serious consideration should be given to relocating this data to a faster and more accessible storage device.

3. Packet Type Histogram

The Packet Type Histogram records and summarizes the distribution of each type of packet transmitted on the network. A simple example would be the total number of data packets transiting the network during a specified monitoring period. Gathering data to be utilized in constructing a packet type histogram can be easily accomplished by a central monitoring site. A summary of packet types could

provide network operations personnel with information concerning the amount of 'overhead' data in relation to the amount of 'pure' data being transmitted. Additionally, it may be found that there exists certain times when the network may be carrying a disproportionate amount of overhead data as a result of component failure, excessive measurement, or excessive monitoring requirements.

4. Data Packet Size Histogram

This histogram records the number and proportion of data packets that fall into a class of specified length. These classes can be either preset or operator defined. For packets of fixed size, the data portion alone may be counted and utilized as the criteria for class inclussion. Variable size packets allow for a strict count of bytes making up the entire packet. The use of a Data Packet Size Histogram can be extremely useful in a network utilizing packets of a fixed length. If the average or mean length of data carried in any one packet is substantially below the carrying capacity of the fixed data field, consideration should be given to reducing the size of the fixed data field. This will reduce the amount of 'excess baggage' being carried by packets throughout the network. Likewise, if packet data fields are full a good portion of the time, or nearly so, consideration should be given to increasing the size of the data field.

5. Throughput-Utilization Distribution

The Throughput-Utilization Distribution indicates the flow of bytes on the network. Both information (data) bytes and total bytes are measured. Information bytes do not include header bytes, or unacknowledged data packets. Additionally, bytes involved in collisions are not counted. Using this approach, total channel throughput, channel

utilization, information throughput and information utilization can be determined for the network . In this way, a true picture of the beneficial usage of the network can be obtained. Collecting the measurements required for the creation of this report is a simple task which can be performed by the central monitoring site.

6. Packet Interarrival Time Histogram

The Packet Interarrival Time Histogram indicates the number of packet interarrival times which fall into particular time classes. An interarrival time is the time between consecutive carrier (network busy) signals. This measurement can assist in determining how much the network is being used and what percentage of the time the network is idle during a specified monitoring period. If a large percentage of interarrival times fall into a class which records occurences of large interarrival times, then it is safe to conclude that, during the monitoring period in question, the network was not highly utilized. When taking these measurements from a central monitoring site, consideration should be given to the fact that the recorded interarrival times will be slightly biased due to the propagation delays between the adaptors and the monitoring site. In the high speed environment of a local area network, these delays are seen as being negligible.

7. Channel Acquisition Delay Historgam

The Channel Acquisition Delay Histograam depicts the time spent by adaptors contending for and acquiring the channel. The channel acquisition delay begins when an adaptor becomes ready to transmit a packet and ends when the first bit is transmitted into the channel. Included is all of the time spent deferring due to a busy channel and the time recovering and backing off from one or more collisions.

From these measurements, we can identify for each interface, the number of packets whose deferral times fell into various time classes. When using a CSMA/CD access protocol, it would be appropriate to assume that, under similar conditions, the distributions of channel acquisition delay times for all adaptors should appear very much the same. If there is some variation, this is a good indication that some type of problem exists within a particular adaptor. Additionally, the mean channel acquisition delay time and it's associated standard deviation can be determined from data contained in the histogram. The collection of this data must be accomplished by each individual adaptor. Results of the measurements taken at each interface must then be forwarded to the central monitoring site on demand, or at some prearranged time.

8. Communication Delay Histogram

The Communication Delay Histogram indicates the delays that adaptors incur in communicating packets to their destination. Theoretically, a communication delay begins when an original packet becomes ready for transmission and ends when that packet is received by the destination. By definition, a communication delay excludes the time to generate and communicate an acknowledgment packet back to the original sender. As implemented by the NBSNET, communication delay is measured from the time at which a packet is ready for transmission until the last bit of the packet is transmitted onto the channel. This value is saved until the transmission is acknowledged, at which time a local histogram is updated. From this it can be seen that measurements must be taken by the adaptor and sent to the central monitoring site upon demand or at a predetermined time. With this approach, the communication delay time recorded will not include the time to propogate the signal to the

destination. This is taken into consideration when measuring 'one hop' delay. Although similar to communication delay, 'one hop' delay includes propagation delay time, and the time for the destination to communicate it's acknowledgment back to the source. The delay, communication or 'one hop', measured depends upon the goals and objectives of the measurement operation.

9. Collision Count Histogram

This Histogram tabulates the number of collisions a packet of any type encounters before being transmitted. Interpretation of these statistics provides an indication of the efficiency of a CSMA/CD protocol in allowing interfaces to acquire the channel. Recording of collision information for each specific packet must be accomplished at the local level. Every time a packet is involved in a collision, a counter in the packet header is incremented by one. Upon successful transmission, the number of collisions incurred by the packet prior to transmission is read directly from the packet header by the central monitoring site. Transferring information in this manner to the central monitoring site would require a modification to the packet format proposed in [Ref. 2]. This modification would require the inclusion of a field for the number of collisions experienced by the packet prior to successful transmission. By combining collision count information from throughout the network, the central monitoring site is able to determine the mean number of collisions per packet transmission and it's associated standard deviation for the entire network.

10. Transmission Count Histogram

The Transmission Count Histogram indicates the number of times a packet is transmitted before it is communicated to its destination. A packet is communicated when it is successfully received by the intended destination. A packet may be transmitted but not communicated due to a collision, line noise, or erroneous transmission. The number of times a packet is transmitted before it is communicated can be detected by the central monitoring site. It does this by observing packet sequence numbers and is thus able to recognize the first through the last times a particular packet is transmitted and which transmission is the communication. Through the use of this histogram, we are able to determine the total number of packets transmitted, the total number of packets successfully communicated, the mean number of transmissions prior to successful communication and the associated standard deviation. Under ideal conditions, the number of transmissions per communication is one. In a fully operational network this will probably not be the case, the actual value being dependent upon the load on the system and the current network configuration.

C. CHAPTER SUMMARY

We began this Chapter with an overview of the various network measurement tools. The format of our overview called for defining a specific measurement tool, followed by a discussion of that tool's prominent measurement characteristics. The tools discussed were: cumulative statistics, snapshot statistics, trace statistics, emulation, artificial traffic generators, and a measurement/control center. The topic of measurement tool selection and implementation was then presented. An approach was offered as a means through which measurement tool selection could take place. This

approach requires that the object of the measurement operation be defined, followed by a statement of the goals of the measurement operation. Next, the performance measures that best characterize the area of the network under investigation are selected. From this, the measurement tools most appropriate for use in obtaining the required information from the network are identified and implemented.

Having concluded that it would be infeasible to identify and provide rationale for every measurement that could be taken from a local area network, a discussion concerning the measurements currently being taken on an operational local area computer network was entered into. Ten performance reports implemented on the NBSNET were explained and their relevance to the SPLICE LAN discussed. These reports were: Host Communication Matrix, Group Communication Matrix, Packet Type Histogram, Data Packet Size Histogram, Throughput-Utilization Distribution, Packet Interarrival Time Histogram, Channel Acquisition Delay Histogram, Communication Delay Histogram, Collision Count Histogram, and Transmission Count Histogram.

The question , 'How much of the network traffic should be measured?', was implicitly addressed in our discussion of artificial traffic generators. Basically, two approaches exist. By measuring everything on the network it would be possible to totally reconstruct the original traffic. Some problems exist with this approach. First of all, there would be a prohibitive amount of storage required for the data collected from the network. Secondly, the review and analysis of this information would take an excessive amount of time. Finally, it may be found that adaptors are spending an inappropriate amount of time collecting and processing measurement data.

The second approach to network measurement employs a sampling technique. Here, performance measurements are constructed only from a subset of the total packets transiting the network. Measurements can be randomly taken of the normal packets flowing on the network, or from those packets created explicitly for measurement purposes by an aritificial traffic generator. In the first case, no control is exercised over the packets being transmitted through the network. In the second case, control of the packets is possible. The characteristics and benefits of artificial traffic generators have been previously discussed in this thesis and in [Ref. 12]. Additional justification for the implementation of artifical traffic generators is provided by Tobagi in the statement, "Generally, internal subnet performance is better studied in a controlled traffic environment rather than in a real traffic environment" [Ref. 6: p. 1442].

To obtain a thorough performance analysis of the SPLICE LAN, this researcher feels that network operations personnel must be able to generate known artificial traffic loads on the system. To implement this capability, it is proposed that each adaptor be able to function as an artificial traffic generator. Process activation, deactivation and parameter establishment would be controlled by the central monitoring site. Additionally, it is recommended that, for specified monitoring periods, the network possess the capability to measure every occurence of certain types of events. This capability is required in order to create various matrices and histograms (e.g., Host Communication Matrix, and Packet Type Histogram).

The researcher does not feel there exists an urgent requirement for an emulation capability. The composition of the SPLICE configuration has been established and is reflected in [Ref. 13]. Possibilities for expansion would

seem to be in the area of additional host processing capability. It is the opinion of this researcher that the controlled addition of processing capability will not tax the networks ability to satisfactorally deliver packets. This conclusion is based on, review of a report dealing with the performance evaluation of the Ethernet local computer network [Ref. 14], and on the assumption that there exists enough similarity between the SPLICE LAN and the Ethernet to justify a conclusion of similar performance under increased loading conditions.

It is recommended that the ten measurement reports discussed in this Chapter be adopted as the basis upon which the measurement capability for the SPLICE LAN be established. It is the opinion of this researcher that these reports provide an accurate and fairly comprehensive picture of network performance which can be utilized by operations personnel in managing the network. Additional measurement reports that could augment those already presented would possess the ability to measure response time, processor and line utilization, characters and messages received in error per unit time, average delay, and software queue lengths and buffer counts such as in adaptors and shared resources.

Uses for measurements taken from a computer network include performance analysis, and component failure identification, isolation, and testing. The degree of success achieved in the accomplishment of these tasks is highly dependent upon a comprehensive and accurate measurement facility. To insure this capability continues to be provided throughout the life of the network, it is imperative that the measurement software incorporate a flexible design in order to accomidate expansion of, and modification to the measurement tools.

IV. NETWORK PERFORMANCE ANALYSIS AND COMPONENT FAILURE

In Chapter 2 we presented and discussed various monitoring methodologies. The concept of network measurement was then undertaken in Chapter 3. Using the knowledge imparted by these Chapters, we can now discuss the topics of network performance analysis and component failure handling. Basically, network performance analysis is concerned with evaluating data and statistical reports obtained by the network's measurement function. During this evaluation process, measurements are scrutinized for signs of component failure and inefficient network functioning. Additionally, performance analysis of the network allows us to: adjust network performance parameters in order to 'tune' the network, plan for network growth, and identify bottlenecks at various components throughout the system. For our discussion, the concept of failure has been more broadly defined to include the network's inability to provide timely service to it's users. What this means is that degradation of selected performance measures, such as network throughput, will be classified as a failure within the network.

Initially, a discussion dealing with the question, 'At what time should the performance ayalysis take place?', is entered into. We then look at the function of performance analysis as it pertains to a local area computer network. Finally, a presentation and evaluation of various techniques used in the detection and diagnosis of network component failure is undertaken.

A. PERFORMANCE ANALYSIS TIMING

There are three time frames in which performance analysis can take place. These are on-line, off-line, and instantaneously. Off-line analysis requires the evaluation of performance measurements to take place upon completion of the monitoring period. On-line analysis enables the evaluation to take place during the monitoring period. Evaluating data at this time implys a delay between the generation of the measurements, their analysis, and subsequent actions taken as a result of this analysis. Instantaneous analysis is accomplished through the use of dynamic control programs. These programs provide for the immediate ayalysis of data and statistical reports, followed by any corrective action that may be required.

1. Off-Line Analysis

Off-line analysis implys that the records generated by the monitoring system are placed in mass storage for future analysis. Performance analysis is accomplished in this way by the NBSNET [Ref. 8]. Delay in corrective action initiation due to off-line analysis experienced by the NBSNET was 5-10 minutes [Ref. 8: p. 726]. Implementing this 'method' of performance analysis provides the analyst with the ability to obtain an overall picture of network performance before making any otherwise rash parameter adjustments. This method also allows a more indepth analysis of the performance measurements through the use of off-line testing and evaluation programs. An additional reason for the use of off-line analysis is based upon the speed of the LAN. The high rate at which packets are transmitted means that there is only a small amount of time to simultaneously assimilate the data and create statistical reports upon which to act. The major problem associated with off-line analysis is

it's lack of responsiveness. As a result of the speed of a LAN, the environment which was recorded during the monitoring period may not exist upon completion of the analysis. Therefore, any adjustments to the parameters based upon the analysis may no longer be applicable to the current environment.

2. On-Line Analysis

On-line performance analysis enables network operators to capitalize on the benefits offered by a real time computational environment. Although the degree varies, on-line performance analysis is currently practiced on the Los Alamos Integrated Communications Network [Ref. 21], and the Lawrence Livermore National Laboratory Octopus Network [Ref. 24]. Additionally, the Codex Distributed Network Control Systems 200 and 300 utilize an on-line approach to performance analysis [Ref. 25]. This 'method' of analysis provides for: a more immediate detection, diagnosis and correction of network failure, a greater utilization of advanced graphic capabilities for monitoring the network, and an increased use of decision support capabilities which can provide the operator with suggested courses of action and adjustments to network performance parameters. Two main problems exist with this approach. First, human intervention is still required for the adjustment of parameters inorder to modify specific network performance measures. And second, there remains considerable delay, with respect to the speed of a local area computer network, between the capture of network performance measurements and subsequent action to effect their values.

3. Instantaneous Analysis

It is the researchers opinion that the implementation of an instantaneous performance analysis capability could theoretically optimize the efficiency and effectiveness in which network evaluation is conducted. Networks that have implemented or plan to implement an instantaneous performance analysis capability are the Ethernet [Ref. 14: p. 717], and the Defense Data Network [Ref. 19: p. 4]. This technique allows network operations personnel to establish ranges within which performance measures may vary. If measures for which ranges have been established breech these predefined limits during normal network operations, an interrupt can be generated which initiates a program designed to bring the value of the performance measurement back within the prescribed range. In this way we are taking maximum advantage of the computers ability to process information almost instantaneously and thereby providing an immediate response to current network conditions. Instantaneous analysis and dynamic control of a network is no longer just a theoretical concept. Advanced installations can now offer significantly simplified or even automatic intervention such as automatic restarts, automatic remote-site monitoring, and electronic reconfiguraton [Ref. 20: p. 10]. The major problem with this technique is that there exists a loss of explicit control of the network by operations personnel as the monitoring, performance analysis, and parameter adjustment become more automated. Additionally, unless steps are taken to insure otherwise, the automation of these procedures may well deprive network operators of information concerning just what is happening inside the network.

B. LAN PERFORMANCE ANALYSIS

Performance is the property of a system that: it works, it is responsive, and that it is available [Ref. 15: p. 4]. Given this, our performance analysis technique must enable us to ascertain that these characteristics are accomplished in the most efficient manner possible By implementing a performance analysis capability, we hope to obtain information that: can be utilized to increase system responsiveness and reliability, will assist in capacity planning, and reduce network operating costs. Additionally, tracking network performance will assist operators in pinpointing more precisely the nature of a failure, thereby helping to correct it quicker and reduce component downtime. This section includes the identification of those performance metrics that have been selected by the researcher as those which can be most effectively utilized in the analysis of SPLICE LAN performance. Additionally, a discussion of performance parameter identification and selection is undertaken.

1. Performance Measure Utilization

Utilizing the information provided by the ten reports explained in Chapter 3, we are able to effectively analyze the performance of the LAN. The question that must be answered now is, 'What measures do we look at, and how do we combine them to insure a complete and accurate representation of the network's performance is obtained?'.

The selection and combination of measurements for the purpose of network performance analysis is based upon the goals and objectives of the pending evaluation. Our emphasis will be on using performance analysis to assist in component failure detection and in improving the operational functioning of the network. For this to occur, acceptable

ranges for critical performance measures under specific network loading conditions and configurations must be established. These ranges may be established and kept in tables by using analytical models to dynamically determine these ranges at defined intervals for use in comparison against actual measured performance. Values of critical performance measurements which do not fall within established limits should cause an interrupt to be generated which inturn initiates some form of remedial action on the part of the system. Finally, in order to maintain explicit control of the system, network operations personnel must be given the ability to establish and set the ranges for these criteria, and predefine certain values taken from the network as critical when they occur, such that the occurrence will be brought immediately to their attention.

Many possible combinations of performance measurements exist. Metcalfe and Boggs [Ref. 18: p. 401], utilized the criteria of: acquisition probability (the probability that exactly one station attempts a transmission and acquires the channel), wait time (the mean time a packet must wait before successfully acquiring the channel) and channel efficiency (that fraction of time the channel is carrying good packets) to evaluate the performance of the Ethernet. This approach is more of an experimental nature and, in the opinion of the researcher, seems to be limited in its usefulness in an operational environment.

Another possible combination was suggested by Tobagi in a presentation at the Naval Postgraduate School in Monterey California on the 21st of October 1982. One of the topics addressed in that presentation dealt with identification and utilization of performance measures for a local area computer network. These measures were: bandwidth utilization, system capacity utilization, and message delay. By breaking these down into more specific monitoring areas,

we are provided with the ability to obtain a comprehensive picture of network performance. These disaggregated performance measures fall into two categories. The first category provides for the evaluation of the networks communication capability and includes as criteria: throughput, response time, and file transfer rate. The second catagory provides for the evaluation of resource utilization throughout the network and includes: processor utiliization, buffer utilization, and line utilization [Ref. 16: p. 48]. The combination of these measurements, together with control of the parameters which effect their values, enable network operations personnel or dynamic control programs to detect degrading network performance and take appropriate corrective action.

2. Performance Parameter Selection

Following the selection of appropriate network performance measures, parameters must be identified which can be adjusted in order to affect the values taken on by these measurements. These parameters and their associated values should be chosen on the basis of existing analytical and simulation results as well as previous experiments carried out in varied traffic conditions [Ref. 22]. The researcher feels that once the parameters that affect the value of a specific performance measure have been identified, they should be prioritized. This prioritization should be based upon the parameters effect on the performance measurement for a given system configuration. This suggests that there may exist different prioritizations of the parameters for different configurations of the network. One possible prioritization scheme might call for the adjustment of those parameters first which have the greatest effect on the value of the performance measurement. An important fact that should be considered when selecting,

prioritizing, and adjusting parameters is that the majority of performance measurements and parameters are interrelated. For example, an adjustment made to increase throughput, such as increasing the size of the packet data field, will also effect the delay experienced by the network user.

Finally, there are many parameters which can affect one performance measurement. Likewise, the adjustment of one parameter is capable of affecting many performance measures. This being the case, it would be extremely difficult at this time to attempt a listing of all those parameters which affect the values of the performance measures we presented above. Rather, these would be more accurately identified through the use of simulation, modeling, and experimentation. In general, one cannot identify a single tunable parameter which directly affects one specific performance characterizing measure. Instead, one can identify the two sets (parameters and measures), and through experimentaton, define their intersections [Ref. 26: p. 1].

C. COMPONENT FAILURE

Along with managing the local area network-longhaul network interface, component failure detection and diagnosis is seen as the most important function of network management. The ability to provide users with a responsive, available network is of primary importance. To do this, we must be able to quickly detect and diagnos network failures. Having this capability will allow the system to immediatly initiate appropriate recovery procedures and restore full service to it's users. This section will address the topics of component failure detection, failure diagnosis, and reporting of the failure throughout the network.

1. Failure Detection

A failure detection function should enable network operators to recognize operating and configuration problems immediately so they can intervene in a timely fashion to correct them [Ref. 20: p. 10]. Not only do we want to be made aware of catastrophic failures, but also of gradually failing conditions. It is a well designed performance analysis capability that enables us to be aware of the latter.

Component failure detection within a local computer network can occur in many ways. Probably the most simple being a face to face encounter between a user and network operator, the subject of discussion being either an inoperable component or unsatisfactory network service. A phone call from a remote user is another method of detection. We can also see a network operator laboriously reviewing system statistic reports for signs of degrading performance. From these passive monitoring techniques which required extensive operator intervention, the emphasis has shifted and is now on automatic alerts based on equipment failures, and in more sophisticated applications, also on user-defined limits on such items as transmission volumes and response time [Ref. 20: p. 10]. Implementation of an automatic failure detection capability is currently being planned for the Defense Data Network [Ref. 19: p. 7].

It is not the researcher's intent to suggest that all, or any subset of the detection methods to be presented below should be automated. Rather, the author's approach will be to identify and discuss possible techniques of failure detection, understanding that their implementation could take a variety of forms.

a. Maintenance Detection

Failures can be identified through normal network maintenance activities. These activities may be under operator or program control and may occur at predefined intervals or on an as needed basis. For example, in the process of updating the configuration data base, the need may arise to poll all components within the network. No resonse from a particular element may indicate the existance of a failure. Additionally, the failure to receive a required maintenance or status report from a component is another indication that a problem may exist. Testing the operation of the network utilizing artificial traffic generation may also lead to the discovery of network inefficiencies. Finally, the use of watchdog packets [Ref. 8: p. 727] to verify active and inactive components is also a viable tool that can be used in identifying failing elements.

b. Performance Analysis Detection

It is the researcher's opinion that the major benefit to be gained from the performance analysis of a LAN, is the added capability it gives network operations personnel in detecting failed components. Status reports generated by individual components, and those created by the central monitoring site can be reviewed for: changes of state, obvious trends, and erratic component performance. Additionally, component error counts can be reviewed for degrading conditions. Two systems which use approaches similar to these are SNA [Ref. 27: p. 12], and the Arpanet NCC [Ref. 23: p. 6-6]. In SNA, Record Maintenance Statistics are generated periodically and sent to the control point where they are logged and scanned to detect degrading component performance. In the Arpanet, IMP's

examine their own status and send reports to the NCC every minute. Finally, by monitoring the availability of a component, which is defined as the mean time between failure (MTFB) divided by (the MTBF plus the mean time to repair (MTTR)), we are able to detect a very gradual degradation of that component's ability to perform it's function over an extended period of time.

c. Localized Detection

Detection of a failure within a component can be accomplished by the component itself, assuming the failure is not a catastrophic one. A trap mechanism within an adaptor or component interface is a 'device' which is activated whenever a certain hardware failure occures or a block of code is executed. This mechanism not only detects the problem, but can used to initiate some type of diagnostic or corrective action. Hardware devices are also used for problem detection at local levels. The Arpanet IMP hardware is capable of automatically detecting power failures [Ref. 23: p. 6-5], while the Ethernet employs a watchdog timer which disconnects the transceiver from the channel if it starts acting suspiciously [Ref. 18: p. 20]. One final method of local failure detection is accomplished by establishing a maximum number of retrys for a packet transmission. After a maximum of 15 retrys to transmit a packet, a transmitter on the Ethernet gives up and reports the failure condition [Ref. 17: p. 20].

Detecting the failure of an adaptor's attached component and any peripherals associated with it is also a requirement. Detecting the failure of an adaptors attached component can be accomplished through the use of an inactivity timer. The purpose of this timer is to signal the possibility that the attached component may have failed. Once the timer runs down, action is initiated to verify the

status of the component. If test results indicate that the component is down, the central monitoring site is notified. Finally, it is assumed that failure detection of peripherals attached to the network component will be accomplished by that component. However, the requirement exists that the status of these peripherals be accessible to local failure detection routines inorder that the central monitoring site may be kept aware of their condition.

d. Neighbor Detection

If a node experiences a catastrophic failure without being able to notify the central monitoring site of the impending doom, then we must have a method by which this failure can be detected and the central monitoring site notified. At the local level (.i.e. without assistance from the central monitor) there exist 2 possibilities, both of which are based on the assumption that the failed node was involved in a session when the failure occured, or, that some other node will attempt to initiate a session with the failed node within a reasonable amount of time after the failure. Assuming the node in question is involved in a session, there exist two methods of detection. The first method involves the maximum number of times a packet will be transmitted without receiving an acknowledgement. If, during a session, a packet is successfully transmitted the maximum number of times without receiving an acknowledgement, then the transmitting station can assume the destination is down and notify the central monitoring site. Similarly, if the destination node stops receiving packets from the source node without getting an end of message indication, it can assume the source has failed and notify the monitoring site. Finally, the technique based on nonreceipt of acknowledgements can also be used when one station is attempting to establish a session with another station.

2. Failure Diagnosis

Once a failure has been detected, it must be located, and it's cause determined. These are the primary objectives of a failure diagnosis function. This function may be automated such that the detection of a failure initiates a program which perfroms various diagnostic routines in support of the accomplishment of these objectives. The Defense Data Network utilizes an approach similar to this. As planned, the DDN Monitoring Centers will be capable of automatically monitoring network elements to identify, isolate, and sometimes correct problems without specialized maintenance personnel involvement.

When designing a set of diagnostic tools it should be noted that, for some diagnostic tests and routines, monitoring and normal data traffic flow may be suspended. Assuming this to be the rule rather than the exception, diagnostic tools should be developed accordingly. In the following sections we will identify and discuss a number of these tools.

a. Tests and Traps

Individual diagnostic programs can be utilized to initiate specific tests in areas of the failed component. Additionally, traps can be utilized to activate these programs once a failure has been detected. Tests conducted on the component might include checking all physical connections the component has with other devices and comparing a block of code in the cojmponent with an imagae of what that code should be.

b. Interface Looping

A good diagnostic tool for a network interface is the ability of a node to send packets to itself. In giving a node the ability to transmit and simultaneously receive the transmitted packets, we are able to obtain complete verification of the network interface. This is where our artificial traffic generator comes into use. We can generate a stream of packets with known content, and size and arrival distributions. By checking the returning traffic against what was just generated, we can identify any problems which may exist in our network interface.

c. Dynamic Diagnostic Tool (DDT)

The use of a Dynamic Diagnostic (or Debugging) Tool was introduced by the Arpanet NCC [Ref. 23: p. 6-5]. The DDT is a set of software programs which are utilized in an effort to diagnos the cause of a component failure. The DDT may be local to, or transmitted to the machine associated with the failed component. DDT can be used to perform a number of tests and operations geared towards determining the cause of a component failure, these include: the examination and modification of a specific word in memory, clearing an entire block of memory, searching memory for a particular stored value, examining the contents of specific buffers and modifying their contents, measurement of a device's realtime clock, and implanting traps and interrupt handlers in a device suspected of having software or hardware problems.

d. Dump and Load

If all other diagnostic methods fail to determine the cause of a failure, one final course of action exists. The entire contents of main memory existing within

the component at the time of failure is dumped to off-line storage where additional diagnosis and analysis can be conducted. Simultaneously, a new copy of the appropriate software is loaded into the component. If this procedure still fails to correct the problem or bring the device back on-line, it can be assumed, with a high degree of certainty, that a hardware problem exists and contact of appropriate vendor personnel is in order.

3. Failure Notification

We now address the question, 'Who is notified and what data bases are updated upon the detection of a failed component?'. Assuming detection and diagnosis were accomplished by a distributed component (relative to the monitoring site) in the network, the central monitoring site should be the first entity notified of the failure. Realistically, notification of the various entities to be identified below, could happen simultaneously, or nearly so. It would then be the responsibility of the central monitoring site to notify additional entities and to update the appropriate data bases. These data bases are updated by the central monitoring site in basically two ways, either by operatons personnel or by a program which automatically makes entries into the appropriate data bases upon receipt of failure alert messages. The data bases that must be updated include: the configuration data base, the problem management data base, and a historical data base which is utilized as a means through which the evolution of the network can be tracked.

There are a number of additional entities which must also be notified upon the detection of a failed component. To begin with, if monitoring site personnel were unable to restore the failed component, then the appropriate vendor must be contacted. The rest of the LAN will be notified of

the failure by the problem management data base or configuration management data base when they log onto the network, or when they attempt to utilize the resources normally provided by that component. Users attempting to utilize the resources of the failed component from a geographically dispersed site through the DDN will be notified of the failure in a manner analogous to local users once they have made contact with the LAN. Finally, those members of the operations staff who may be in the process of conducting any type of experiments or monitoring activities which include the failed component must be explicitly notified of the configuration change.

D. CHAPTER SUMMARY

We began this Chapter with a discussion of the possible time frames in which network performance analysis could occure. Those discussed were: off-line, on-line, and instantaneous analysis. The topic of local area network performance analysis was then entered into. In this section we discussed performance measure utilization and performance parameter selection. A presentation of various methods of component failure detection and diagnosis concluded the body of the Chapter.

It is the researcher's opinion that a SPLICE LAN could benefit from each type of analysis. Instantaneous analysis could be utilized to evaluate and effect the performance of the network layer protocol and below. This would reduce management overhead in that personnel would not be needed to constantly monitor network status via a CRT, or to review and analyze printouts reflecting the networks condition. For example, adjustments made to increase throughput during times of network congestion, such as modifying our backoff technique, would be accomplished by a program rather than

requiring human intervention. Overhead costs associated with the running of these programs would have to be compared against the costs incurred by non-automated and semi-automatic procedures in order that efficient and cost effective functional implementation is achieved. An on-line analysis capability would give the operator a window through which the functioning of the network could be observed. Through the use of some sort of decision support system, the operator could obtain assistance, possibly in the form of suggested action or in adjusting parameters which effect global performance measures. Off-line analysis would provide operators with the ability to analyze the performance of the network in an environment seperate from the system. This 'method' would remove any pressure that might be experienced by the operator when attempting to analyze performance while on-line.

The performance measures suggested by the author for the SPLICE LAN are seperated into two catagories. The first catagory provides for the evaluation of the network's communication capability. The second catagory includes measures which can be used to evaluate resource utilization throughout the network. Each of these was described in detail earlier in the Chapter. Ranges for these measures should be determined dynamically during network operation, however, network operations personnel must be able to override dynamic range establishment and set their own range values as needed. Numerous parameters exist which can be utilized to effect the values of these performance measures. Rather than proposing a list of tunable parameters, that, by it's very nature would be incomplete, the author offers three suggestions for their identification and utilization. These parameters and their associated values should be established on the basis of existing analytical models and simulation results as well as operational experimentation.

Once identified, these parameters should be prioritized in a manner which reflects their effect on specific performance measurements. Finally, the fact that adjustment of one parameter may effect the value of more than one performance measure must be considered in selection and implementation of the parameter.

Our discussion of a failure detection and diagnosis capability as part of the SPLICE LAN will emphasis the limiting of these capabilities possessed by components distributed throughout the network. The failure detection capability of a distributed component is limited to the identification of those failures which cannot be detected by the central monitoring site. The diagnosis capability is also to be similarly restricted. It is the researcher's opinion that this approach will reduce diagnostic software duplication throughout the network, eliminate maintenance on distributed diagnostic tools, and provide for more central control of failure analysis and problem management. Upon detecting a failure, the component will send some form of problem alert message to the central monitoring site. From that point, the actions taken by the monitoring site are identical to those that it would take if it had detected the failure. Since we have limited the detection and diagnosis capability of the distributed components, conversely, we must increase those of the central monitoring site. It is felt that periodic status reports such as those described in Chapter 3 should be sent to the central monitoring site on a regular basis . There they can be analyzed for possible signs of component failure and system degradation. In addition to those capabilities alluded to above, the monitoring site must be able to direct the transmission of status reports from distributed network components to itself. It must possess a diagnostic tool that can be utilized throughout the network to isolate and identify failures in a

manner similar to that of the DDT employed by the Arpanet Network Control Center. More details concerning the functioning of a central monitoring site will be discussed in Chapter 6. It is sufficient to conclude at this time that, by centralizing the majority of the network's failure detection and diagnositic capabilities, we are increasing control of the failure handling procedures of the network, reducing software duplication and maintenance, and minimizing costs associated with the implementation of a failure detection and diagnostic function.

V. MANAGING THE LAN/DDN INTERFACE

As stated in Chapter 4, along with failure identification and correction, the most important function of LAN management is the monitoring and control of the local area network to long haul network interface. This function is primarily concerned with regulating the flow of packets between the networks and any other tasks which support it's accomplishment. In this Chapter, our emphasis will be on identifying and discussing the managerial aspects associated with the interconnection of two networks.

A fundamental aspect of internetwork communication is the establishment of agreed upon conventions. Communicating entities must share some physical transmission medium and they must use common conventions or agreed upon translation methods [Ref. 29: p. 1392]. This required commonality can be achieved in a number of ways. Protocols of one net can be translated into those of another, or, common protocols could be defined. Another method through which commonality may be achieved calls for conversion to a standard interface by all networks. It is the researcher's opinion that the connection of long haul networks to local area networks does not lend itself to the establishment of common protocols that would be efficient for both networks. Additionally, the benefit to be derived from converting to a standard interface is only realized if a network is connected to more than one other network. If connected to only one network, utilizing a standard interface would require two protocol translations. Network A's protocol would require translation into a standard interface protocol which would then require translation into network B's protocol upon arriving at the connected network. Whereas, the use of protocol

translation would only require the conversion of A's protocol to B's, or vice versa, depending upon the direction of packet flow. Therefore, we will consider the issue of managing the interface between two networks from the standpoint of protocol conversion, rather than from common protocol or standard interface establishment.

There are many differences which exist between networks that must be resolved, those that will be covered in detail in this Chapter include: naming and addressing, flow and congestion control, packet size, and access control. Additional areas to be discussed will encompass gateway configuration, internetwork accounting, and dispersal of network status information.

A. GATEWAY CONFIGURATION

In an effort to set the stage for a discussion dealing with the management of an interface between two networks, it is felt that an understanding of possible gateway configurations, or levels of interconnection as dubbed by Cerf [Ref. 29: p. 1392], will prove beneficial. There are a variety of different ways in which the gateway between two packet switched networks may be configured [Ref. 28: p. 4-49]. We will briefly describe each one and discuss why it should, or why it should not be considered for the SPLICE network. Finally, for explanitory purposes, we will select a configuration and use it as an example throughout the Chapter.

Utilizing a common host is a simple and very straight forward approach that can be used to connect two networks. This method connects two networks through a host that is attached to the two networks. This configuration can be ruled out immediately from consideration for the SPLICE network. This is because the entire SPLICE program is based

upon relieving the host(s) of communication responsibilities. To burden the host computer with anything but the processing of application programs would be entirely against the SPLICE concept.

Another approach to interconnecting packet switching networks would be to have a switching node which is common to both of them. This method must also be ruled out from consideration. First of all, the LAN does not possess a switching node. An attempt might be made to combine the functions of a DDN switching node and the LAN front end processor(FEP). Although a technically feasible solution, the drawbacks are major and numerous.

An internode device can be used as a separate entity to perform only gateway functions between each of the networks to be interconnected. This gateway is normally designed to appear as a special host to each network. This approach provides the most acceptable alternative, however it is the author's opinion that the requirement for additional hardware to perform the interconnection of two networks is not supportive of the SPLICE concept.

The final possibility for a gateway configuration utilizes the existing capabilities of a DDN switching node (IMP), and the local area network FEP. This configuration is called the "two half-gateway". In the "two half-gateway" approach, a gateway is composed of two halves, each associated with it's own network. Each half-gateway would only be responsible for translating between the internal packet format of it's own network and some common internetwork format. The number and different types of networks the DDN ties into will dictate whether or not an approach of this nature is optimum. For the time being, no standard internetwork format has been proposed. This being the case, a slight modification to this approach should make it usable and efficient for connecting a SPLICE local area network to

the DDN. This change would require a conversion from the internal protocol(s) of the local area network to the protocol(s) of the Defense Data Network and vice versa, depending on the direction the packets are flowing.

For the remainder of this Chapter, we will utilize the "two half-gateway" as our basis for explaining the differences that must be overcome when connecting two networks, and the functions which must be accomplished by the gateway. For our discussion, it may help to picture one half of the gateway implemented in the LAN FEP, and the other half of the gateway resident in a DDN switching node which, in conjuction with the LAN FEP, allows communication between the two networks to be achieved. Finally, a number of assumptions have been made, which are felt will add clarification to concepts discussed, and provide a basis upon which analysis can be conducted and proposals made.

1) The LAN cannot affect the speed at which packets transit the DDN.
2) The LAN FEP cannot increase the rate at which packets are sent to it from the switching node past the maximum transmission rate of that node.
3) The switching node that the LAN ties into may also act as an IMP through which other hosts, not part of the LAN, access the DDN.
4) Error control, flow control, and duplicate packet detection is provided for communication between the LAN FEP and the DDN switching node by one of the network access protocols supported by the DDN. In this situation, the switching node merely views the front end processor as another host.

B. PACKET SIZING

The problem of differences in packet size is basically one of coping with the fragmentation that must inevitably occur when the two interconnected networks employ different internal maximum packet sizes [Ref. 28: p. 4-49]. Two situations may exist, one is when the maximum packet size for the LAN is greater than that of the long haul network (LHN), the other being when the maximum packet size for the long haul network is greater than the maximum packet size for the local area network.

The first case, when LAN maximum packet size is greater than the long haul network maximum packet size, can be handled in one of three ways. First, if the packet to be transmitted from the LAN to LHN is smaller than the LHN maximum packet size by at least the number of additional overhead bytes that will be added on by the packet switching node once the packet reaches the DDN, then the packet requires no size modification before being sent to the switching node. Second, if the packet to be transmitted is larger than the LHN maximum packet size, we may fragment the packet appropriately in the FEP. Each packet would be fragmented such that the new packets would be smaller than the maximum packet size for the LHN even after the overhead bytes were added by the LHN switching node. A number of problems exist with this approach which include, a requirement for increased software capabilities at the FEP, additional delay experienced by packets wanting to leave the network, and the possibility that resequencing of all the packets making up the message being sent may be required due to the insertion of a "new" packet into the sequential series of packets that have been transmitted from somewhere in the LAN. And finally, if the packet to be transmitted is larger than the LHN maximum packet size, we may just go

ahead and send the packet to the switching node. We are able to do this because the DOD Standard Internet Protocol, which will be implemented by the Defense Data Network, provides for a fragmentation/reassembly service. It is envisioned that the "over-sized" packet would be fragmented with each piece being sent to the destination switching node where the fragments would be reassembled back into the "over-sized" packet.

In addressing the second case, where LHN maximum packet size is greater than LAN maximum packet size, we assume that the fragmentation of a smaller LAN packet to help fill up a partially filled larger LHN packet will not occur. In this situation, the main concern of the LAN is that it might receive a packet from the DDN which is larger than it's maximum packet size. This being the case, the LAN FEP must possess the capability to fragment the larger packet into packets suitable for transmission on the local area network.

C. CONGESTION CONTROL

Assuming probabilistic message generation and fixed capacity in network components, overload would be inevitable without certain mechanisms to stop, slow down or absorb the rate of message arrival. The basic tool utilized in the accomplishment of these tasks is congestion control. Congestion control can be defined as a procedure whereby distributed network resources, such as channel bandwidth, buffer capacity, and CPU capacity are protected from over subscription by all sources of network traffic [Ref. 29: p. 1400]. Congestion is most likely to be visible at a gateway connecting a local area network to a long haul network. In some cases, the transmission rates of LAN's might exceed those of long haul networks by factors of 30-100 or more [Ref. 29: p. 1400]. There are basically two schools of

thought when it comes to dealing with the problem of congestion control. There are those who advocate rigidly controlling the input of packets into a network and explicitly rule out the discarding of packets as a means of congestion control. And conversly, there are some who promote the dropping of packets as the sole means of controlling congestion [Ref. 29: p. 1400]. We will look at congestion and flow control at the interface between two networks from both of these viewpoints. It is the author's intention to propose and discuss techniques of congestion and flow control for receipt ot packets from the LAN and for receipt of packets from the DDN by the half of the gateway resident in the LAN FEP.

1. LAN to LHN Packet Control

The author has concluded that there exist numerous methods of congestion control, many of which have yet to be identified. The discussion which follows includes the presentation of three possible methods of gateway congestion and flow control. These methods deal with the handling of packets received from the LAN by the front end processor destined for the DDN.

The simplist method of congestion control provides for the immediate transmission of packets to the DDN. If the gateway portion of the FEP, in conjunction with it's associated switching node, is able to successfully transmit packets to the DDN faster than they arrive from the LAN, then we can assume the requirement for congestion and flow control is minimal in that direction. However, the author has concluded that this is rarely the case. This approach would inevitably lead to the loss of packets due to the gateways inability to transmit them at a rate comparable to that of LAN. Recovery/retransmission of those 'lost' packets to the gateway would be left to the lower level protocols.

Another method through which congestion control at the FEP could be accomplished would be through the addition of buffers. Packets flowing in from the LAN could be queued in a buffer for subsequent transmission to the long haul network. Once this buffer becomes full, packets could be discarded as in the first method or a signal of some type could be sent throughout the network indicating that the DDN output buffer was full. Receipt of this message would also imply that no internetwork traffic should be sent until a message is received from the gateway indicating that the buffer is empty and internetwork traffic transmission can be resumed.

This technique could also be employed with two buffers. Once one buffer was full, it would be disabled from receiving additional packets while transmission took place. Simultaneoulsy, the second buffer could be filled and it's contents transmitted when the first buffer became empty. While the second buffer's contents were being transmitted to the DDN, the first buffer would be receiving packets from the LAN. This alternating technique could be employed with N buffers , but this would be at the expense of loosing N buffers worth of memory space in the FEP. This being the case, a limit to the buffer space allocated to internetwork traffic would have to be established. With this limited buffer space, there still exists the possibility that all buffers may become full simultaneously. This would require incoming packets to be discarded or, notification throughout the network that buffers are full and internetwork packet transmission is disabled.

A final method by which the flow of traffic from the LAN to the LHN can be controlled is through the use of external storage areas. This technique is very similar to the buffering methods presented above. Buffers are utilized in the same fashion but, when they become full, rather than

discarding packets or notifying the network of the buffers state , all incoming packets are directed to external storage areas. When the buffers begin to empty, packets currently being stored are directed to the output buffers on a FIFO basis. This procedure reduces congestion on the LAN by not requiring the continual retransmission of packets not previously accepted by the gateway. Addionally, it eliminates the need for distributed components to be able to recognize a "DDN buffers full message" and carry out the internetwork packet restricting action necessitated by it's receipt.

2. LHN to LAN Packet Control

As previoulsy stated, we are assuming that the flow of packets between the FEP half of the gateway and switching node half of the gateway is controlled by the network access protocols supported by the DDN. This being the case, our discussion is restricted to answering questions such as: 'Should we transmit each packet immediately onto the LAN upon it's receipt from the DDN?', or 'Should we employ some buffering techniques, accumulating some packets before transmitting them onto the LAN?'.

It is the author's opinion that the trickling of packets onto the network one at a time does not efficiently utilize the capabilities of a 10 Mbit/sec LAN. This method reduces network throughput, and requires adaptors and components to "wait" longer between internetwork packet arrivals. By storing the internetwork packets in buffers or dedicated external storage areas, we are able to transmit packets onto to the LAN in bursts. These transmissins can occure after an entire message is received or after a certain number of packets have accumulated in the buffers or external storage areas.

D. ADDRESSING AND NAMING

Whenever any two devices must communicate with each other and they are not directly connected (i.e. a processor on one network communicating with a processor on another network), the question of addressing the proper recipient becomes a major consideration. Addressing across network boundaries requires either a standard network numbering scheme or a means of address translation in the gateway [Ref. 28: p. 4-49]. It is known that the DDN will connect with existing networks as well as the SPLICE local area networks. It is the author's opinion that this in itself is sufficient to justify the establishment of a standard numbering scheme. This will therefore be the premise upon which our discussion will be based.

Many different possible internetwork addressing schemes exist. The CCITT X.121 addressing strategy is based on the telephone network system. This technique allows up to 14 digits per address. The first 4 digits are a destination network identifier code (DNIC), followed by the remaining digits which may be used to implement a hierarchical addressing structure [Ref. 29 p. 1400]. The DARPA Internet has implemented a common address format across all networks it connects [Ref. 30: p. 114]. The Internet address length is fixed at 32 bits. These bits contain the address of a particular network, and the address of a host within that network. A further disaggregation of this concept might call for an address field which contained a network address, the address of a packet switching/gateway node within that network, and the address of a host accessible through that node. We will utilize the addressing technique implemented in the Internet as the basis for the remainder of our discussion.

In order to manage, control, and support communications among components distributed throughout two or more networks, a means must exist for explicitly identifying the components involved in the communication. This could be accomplished by utilizeing one of the addressing strategies presented above. In implementing this strategy, rather than requiring the user to be aware of the structure of the network in which the destination host resides, a naming convention could be established which relieves him of indicating the actual address of the desired host. A naming convention can also be established for identifying the network to be accessed rather than requiring a specific address to be provided.

Assuming an operator may now use names to identify both the destination network and the host within that network, the task of converting these to actual network addresses must be considered. Translation of the network name to a specific network address will be accomplished by the switching node through which a SPLICE LAN is connected to the Defense Data Network. Currently, nodes attached to the DDN may be known by as many as four different names [Ref. 31: p. 111]. The translation of a local host name to its associated address and vice versa, could be accomplished by the switching node. The author does not support this approach for the major reason that the switching node will most probably be connecting other networks and/or hosts to the DDN. For it to possibly perform these translations would mean a reduction in the node's capability to perform its primary functions of traffic processing, host access, routing, and monitoring and control [Ref. 31: p. 33]. Therefore, local host name translation must be performed at the local level.

This local translation capability could be accomplished at the interface between a distributed component and the bus. This would require the use of additional component resources for the performance of a function which could most efficiently be implemented at a centralized location (i.e. the Front End Processor), rather than at each individual component. By incorporating the local translation capability into the LAN's FEP, we not only reduce redundancy throughout the system, but also facilitate the maintenance of our translation tables. The final issue to be addressed is concerned with the place (source or destination network), at which this translation occurs. Translation of the destinations name can either occur at the source's gateway or at the destination network. By delaying translation of the name to an address until arrival at the destination, we eliminate the requirement for each gateway to possess specific address information about other networks. Similarly, the translation of a process name to a process address would also be accomplished by the destination networks FEP. The half gateway resident in each SPLICE FEP would only be required to maintain a table containing the names and addresses of it's local components and processes. Upon receiving a packet from the DDN, the component and process string names would be compared against entries in the address translation tables. Appropriate addresses would replace the physical node name and process name. The packet would then be ready for transmission onto the local bus.

E. ACCESS CONTROL

Access control is concerned with establishing mechanisms that may be required to prevent some traffic from entering and possibly some traffic from leaving the network. This filtering action is ideally accomplished by the gateway two

networks. Utilizing our model of a "two half-gateway", each half can deal with controlling access to the network that it is connected to. What this means is that our half of the gateway in the LAN FEP can act as a sentry to incoming traffic. As traffic arrives, the "ID" of the packet(s) can be checked against a table containing the "names" of those packets which are authorized to enter the LAN. If a packets "ID" appears on the access list, entry is granted, if not, the sentry may either discard these packets or possibly send them to an access controller [Ref. 29: p. 1401]. The access controller routine can then dynamically enable the flow of the packets into the network after performing certain checks on the packets identity, or, it may decide that these packets are not to be allowed into the network, discard them, and send a suitable 'canned' response to the source of the packet(s) letting it know access was not granted. Alternately, it may inform network operations personnel of the packets that wish to enter the network and request action to be taken.

F. OTHER CONSIDERATIONS

Two additional areas of concern associated with the interconnection of two networks are failure notification and accounting procedures. Assuming that failures for the connected networks are detected, identified, and isolated internally by each network, the question arises 'How is the existance of a failed component within a network communicated to those in other networks who may wish to use that component?'. Assumming that both the LAN configuration data base and problem management data base have both been updated with the current status of any particular failure, the researcher makes the following proposals in response to the previous question. Before packets are let into the local

area network, the half-gateway will be responsible for checking these data bases to insure that the desired destination is operational. If it is, and assuming the access controller has permitted access, the packets are transmitted into the network. If the desired destination is currently inoperative, a response indicating such is returned to the source. Additionally, if a source from another network desires to check the status of an element within a SPLICE LAN, it should have the capability, just as a local user would, of querying either one of these data bases. Also, it is assumed that if the switching node through which a SPLICE LAN is connected to the DDN fails, then the responsibility of reporting the inaccessibility of that particular local area network lies with the DDN Monitoring Center who's juristiction includes the failed node. Similarly, the failure of a LAN FEP which makes a SPLICE LAN configuration inaccessible will be reported to potential network users by the connecting switching node.

It is the researcher's opinion that a SPLICE LAN is seen as just another subscriber to the DDN. This being the case, there seems to be sufficient justification for the establishment of some type of accounting procedures which provide the means through which the flow of packets to and from the DDN can be monitored. Assuming some type of accounting will be conducted by the switching node, the connected LAN could obtain accounting information from it. This does not provide for any type of cross checking of the switching node's accounting capability or accuracy. This then establishes the requirement for some sort of accounting procedures to be established in the LAN's half of the gateway. Currently, public packet switching networks are using procedures which account for subscriber use on the basis of the number of virtual circuits established during the accounting period and the number of packets sent on each

virtual circuit [Ref. 29: p. 1400]. Only slight modifications to certain reports recommended in Chapter 3 would be required to give a SPLICE local area network a similar accounting capability. Finally, and most important of all, is that the accounting mechanisms implemented by the SPLICE LAN's be based upon procedures and units of measure identical, or very similar to those utilized by the DDN.

G. CHAPTER SUMMARY

We began this Chapter with a discussion of various configurations that a gateway between computer networks could assume. The author feels that the "two half-gateway" concept offers the simplest and most effective means of interconnection. The discussion then turned to the problems associated with different maximum packet sizes utilized by the two interconnected networks. We looked at the situations when the LAN maximum packet size was greater than the long haul network maximum packet size and vice versa. In both cases, suggestions were made as to how this problem could be handled. A discussion of flow control and congestion control techniques was then entered into. This problem was approached from two directions. First, controlling the flow of packets into the LAN half-gateway for transmission to the DDN. And second, controlling the flow of packets from the Defense Data Network, through the gateway, into the LAN. The problems of internetwork addressing and component naming were then considered. The author has concluded that the solution to the first problem would be the establishment of a standard internetwork numbering and addressing scheme. The standard offered was of the form, NETWORK ADDRESS/LOCAL HOST ADDRESS. The component naming problem was found to be best handled at two levels. The translation of a network name to a specific

network address would be conducted at the switching node half-gateway, while the translation of a local host name to a local host address would be accomplished by the destination network half-gateway. We then briefly discussed the topic of access control. There, we looked at the role played by an access controller, and attempted to add support for it's implementation in the SPLICE network. Finally, we looked at the need for failure notification and accounting capabilities associated with internetwork traffic.

Exclusive of the interface between a SPLICE LAN and the DDN, the monitoring and management of a SPLICE local area network is predominantly centralized. Special interface functions such as those described in this Chapter require that the control of these functions be distributed to the FEP. Finally, it is the researcher's opinion that the management of the LAN/DDN interface must not only be workable, but must be acceptable from an operational standpoint by the users, and from a technical and logical standpoint by network operators.

VI. LAN CENTRAL MONITORING SITE

The integration of those management tools discussed in Chapters 1-5 is accomplished by the local area network central monitoring site (CMS). It is here where measurements and statistics are collected, performance analysis conducted, diagnostic programs and recovery actions initiated, network utility data bases updated, and where performance parameter adjustment messages originate. This process of managing from a central location minimizes communications and synchronization difficulties, and helps solve problems that may otherwise pass unnoticed [Ref. 33: p. 21].

The author will initially present what he feels to be the mission of a central monitoring site, followed by appropriately supportive objectives. The manning requirements and organizational structure associated with a CMS will then be discussed. From there, a discussion of a network operator's workbench will be entered into. Finally, a discussion of a network operator's responsibilities under both normal and failure conditions will be presented.

A. MISSION OF A LAN MONITORING SITE

The mission of a LAN central monitoring site might be stated as , 'To insure the most efficient and effective use of network resources and to maximize network availability, throughput and responsiveness'. Objectives which support the accomplishment of this mission are:

- Keeping track of the status and configuration of the network.
- Detecting alarm conditions and failed components.
- Carrying out fault isolation and diagnostic tests.

- Contacting appropriate repair personnel and monitoring repair activities.
- Altering the physical and logical network configurations and documenting such alterations.
- Adjusting component performance parameters.
- Generating management reports.
- Supporting test and acceptance activities.
- Provide information needed for planning future network evolution.
- Provide a historical data base against which current and future network performance may be measured.
- Monitor component utilization throughout the network (e.g. host, communication processor, and shared resources utilization).
- Perform a scheduling function for application programs requiring use of the host processors.

The first eight objectives are similar to those containd in the Program Plan for the Defense Data Network. [Ref. 31: p. 142]. The 9th and 10th items are objectives of the Lawerence Livermore Laboratory Octopus Network Monitoring and Measuring Project [Ref. 34: p. 2]. The final two objectives are a product of the author's research.

An analysis of these objectives shows that the tools discussed in this paper are capable of accomplishing the 1st through the 7th and the 11th objectives directly. Although not mentioned as yet, the central monitoring site must be able to support the testing, evaluation and acceptance of new components that are to become part of the local area network. By establishing the data bases described in Chapter 1, we areindirectly supporting the accomplishment of the 9th and 10th objectives. The establishment of data bases which record network performance measures and component utilization statistics would greatly enhance our

ability to meet these objectives. At this time, the researcher does not see a need for the design of a scheduling algorithm as proposed by the final objective. As applications grow, and the number of network users increase, the requirement for establishing a scheduling algorithm for the purpose of efficiently and fairly assigning jobs to host processors, may become very real. An example of where a scheduling algorithm has been implemented is on the Los Alamos Scientific Laboratory Integrated Computer Network [Ref. 35].

B. MANNING AND ORGANIZATION OF A LAN CMS

How many people are required to insure the continued and efficient operation of a SPLICE local area computer network? Should the monitoring site be manned around the clock? What organizational aspects must be considered with the addition of a central monitoring site? These are the questions that will be addressed in this section. During the discussion of each, a possible answer will be recommended by the author.

The manning proposed for the DDN monitoring centers range from four people at the system monitoring center, from one or zero at other centers [Ref. 31: p. 137]. The manning of the NBSNET (a bus oriented local computer network) measurement center calls for 4 full-time and several part-time computer-electrical engineers [Ref. 36: p. 13]. Neither of these manning levels seems appropriate for a SPLICE LAN, the DDN being a much larger long haul network, and the NBSNET manning level reflecting more of an experimental environment. A local area network with a structure very similar to that of a SPLICE LAN is the Hughes Aircraft Company Janet network [Ref. 37]. JANET has centralized the control and monitoring of the network at a single operator position. From this position the operator can issue commands to all

network components, perform testing and performance analysis, detect and diagnose failures, and reconfigure the network. The degree of automation recommended throughout this paper would provide the capabilities required for a 'one person' central monitoring site. If these and other lower level functions are not automated, the possibility exists that an additional operator may be required. The author feels that substantial processing will occur, and that file transfers between Stock Points and Inventory Control Points will take place after normal working hours. For this reason, it is felt that a network operator should be available at anytime processing is in progress. After normal working hours, this position may be filled as a collateral duty (i.e. the individual filling this role may also be responsible for one of the host processors).

In answering the question as to where the monitoring site fits into the organizational picture, one must remember that the CMS can exercise a great deal of control over the network and it's components. This being the case, those individuals comprising the CMS must work directly for the 'Director of the LAN'. We would not want the central monitoring site to come under the control of one of it's users or under the control of one of the staffs associated with a network host. This seems to add more justification for establishing the position of 'Director of the LAN'. This Director could operate out of the central monitoring site. From here, he could manage and control the operations and resources of the local network, formulate network policy, and see to it's implementation.

C. A NETWORK OPERATOR'S WORKBENCH

A network operator's workbench is a single, integrated system containing all the operator's tools in one place. The system must be interactive and, because new analysis packages and models will be continuously developed, possess the characteristics of a programmer's workbench [Ref. 21: p. 4].

Certain hardware assets will enable the operator to better carry out the network management function. The terminal or terminals utilized by the CMS should have a fairly extensive graphics capability. For example, a display of the entire network could be put on the screen with different colors indicating the status of various components. A dedicated printer will be needed for managerial reports, but more importantly, for the recording of failure messages received by the CMS. Adequate direct access storage will also be a necessity. Additionally, an alarm capability for indicating the breeching of established parameter threshholds will be required.

There exists numerous software tools that can be utilized by the network operator. One of the most important is a good DBMS with a complete, user friendly query language. This asset will allow the operator to investigate relationships between performance measurements and associated parameters, and to ask exploratory questions concerning the effect of certain network configurations on performance criteria. The possession of a word processing capability will also assist the operator in the performance of his duties. An additional software asset is the actual process through which the operator interacts with these tools. This interface may be through a Network Operating System as currently planned for the DDN [Ref. 11] and described in [Ref. 32]. Another approach to this problem is to have the

operator interact with the software tools through an application program. The Hughes Aircraft Company has implemented a Network Monitor Program for it's JANET network which runs as an application program on one of the host processors [Ref. 37: p. 96]. The distributed counterpart of this central control program is a 'background' program included as part of the adaptor microcode. It is through these background programs that the CMS receives certain measurements and failure messages. Additional software tools that will be of help to the operator include: an English language set of commands for ease of system operation and network diagnostics, default parameter value establishment if unspecified by the operator, dynamic control programs for adjusting lower level performance paramenters in accordance with network conditions, and finally a system which exists for prompt and accurate collection of any data the user may provide on a problem.

D. OPERATORS ACTIONS: NORMAL CONDITIONS

In the next two sections we will attempt to identify the responsibilities of a network operator under both normal and abnormal conditions. They are presented here in an effort to establish a basic set of responsibilities for all SPLICE LAN operators. This section deals with the operator's responsibilities under normal conditions. It is realized by the author that some of these responsibilities may also pertain to failure conditions. Finally, it is not known which, if any, of the identified responsibilities will be automated. therefore, the discussion of responsibilities will be presented as if the operator had to take some specific action for it's accomplishment.

1. Initialization

Assuming the network operator has just invoked the network management control program, there are certain functions that must be accomplished. The operator must establish a connection with the 'background' program in the adaptor or component interface. Among other things, this will enable him to find out just who is on-line. Once connections are established, the operator can send out instructions to the nodes providing them with guidance as to what measurements to take, when to send them to the CMS, and upcoming maintenance acitvities. Also during this time, the network operator obtains the physical and logical configuration tables for each host and communication processor, which is then stored in the a global network configuration table. During initialization the network operator also sets performance parameter values, establishes alarm threshholds for performance measurements, identifies critical components which he is specifically interested in monitoring and updates the Name/Address Table in the FEP half-gateway.

2. Utility Data Bases

Information obtained from each component about itself and it's associated peripherals is used to update the network configuration data base. This provides the operator with a view of the physical state of the network. Additionally, logical configuration tables can be established for each component and user, which gives them their own 'customized' configuration of the network. Also during this time, problem management, change management and performance analysis data bases may be opened for read/write and checked for items of interest. Finally, the operator needs to communicate with the DDN Monitoring Center who's area of influence the LAN falls within. This interaction

may simply be the transfer of the current DDN status file to the CMS. This file can then be used to assist users attempting internetwork communication.

3. Operator's Displays

The network operator is responsible for monitoring various network status displays and in some cases insuring their availability to users. These status displays are created from data obtained from configuration and problem management data bases in addition to results of performance analysis and component monitoring. As a minimum, the status displays that should exist include: a global network status display, displays for each major component with appropriate operating information, displays for any desired network performance parameters such as throughput and response time, a general information display for informing users of scheduled maintenance, DDN's status and administrative activities, and a display for depicting load information on hosts and communication processors.

4. Normal Management Activities

In addition to those activities mentioned above, the network operator is also responsible for the accomplishment of other normal management activities. He must initiate monitoring periods for the collection of measurement data. Upon the completion of the monitoring period he must: control the transfer of data from the adaptors to the CMS, disable adaptors from taking additional measurements, and clear adaptor memory contents if so required.

Utilizing data gathered, and statistics generated during the monitoring period, the network operator must insure the appropriate data bases are updated. This may include modifications to the configuration, problem management, and performance analysis data bases. Information

obtained during the monitoring period is also to be used by the network operator to identify trends, look for bottlenecks in the network (especially at the DDN/LAN interface), conduct network performance analysis, and prepare network status and utilization reports. While analyzing results of the monitoring period, the operator may become aware of some pending component failure. If so, appropriate action is taken to diagnose and correct the failure. More specific action to be taken upon failure detection will be discussed in the next section.

Other normal management activities include the operators responsibility to test all adaptors failure detection and diagnostic capabilities, the distribution of new software versions, adjustment of network logical and physical configurations, and adjusting performance parameters in order to tune the network. The network operator is also responsible for informing and coordinating with users planned maintenance activities. One final responsibility calls for CMS personnel to be involved in the installation, testing, and acceptance of equipment that is going to become part of the network.

E. OPERATORS ACTIONS: COMPONENT FAILURE

Having utilized the performance analysis, and problem detection and diagnosis techniques presented earlier in this paper, let us assume the network operator has identified a failed component. What then are the procedures that must be followed in order to manage this failure until it's rectification? Assuming the failure is of major significance, such as a down communication processor , one of the first things the operator should do is notify the network of the failed component. Concurrently, configuration tables, the problem management data base, and the Name/Address Table in the FEP

should be updated. Appropriate entries for the problem management and configuration data bases are shown in Appendix A and Appendix B respectfully. Having done this, the operator may utilize some form of DDT as discussed in Chapter 4, in an attempt to correct or further isolate the cause of failure. If this fails, the operator can utilize the information that has been recorded in the network history file to try and 'backup' the processor to a point before the failure occured and attempt a restart. The last chance the operator has to correct the problem is to dump the suspected failure causing software to off-line storage, and reload the system with a fresh copy of the appropriate software. Having exhausted his means of problem correction, the operator is responsible for contacting the appropriate vendor.

During the course of problem identification and correction, it is required that the network remain available for customer use. To do this, the network operator must have the capability of reconfiguring the network, disable processing of local operator requests so that he is in full control of the network, activate and deactivate a components connection to the bus, and transfer functions performed by the failed component to another device capable of performing that function. Although the performance of the network during this time will not be optimum, it will at least be able to support some processing requirements. Upon failure correction, the operator is responsible for bringing the system back to a state of normal operation. This would include updating the appropriate data bases, returning of functional responsiblities as required, and notifying users of the resumption of normal services.

F. CHAPTER SUMMARY

In this Chapter, we began by presenting the mission of a network central monitoring site and the objectives to be met in order to fulfill that mission. A discussion of the manning and structural aspects of a CMS was then entered into. Attention was then focused on the description of a network operators workbench and its' associated tools. Our final discussion dealt with the identification of a network operator's responsibilitirs under both normal and failure conditions.

It is the author's opinion that the mission and objectives presented at the beginning of this Chapter provide a complete and succinct picture of exactly why a network central monitoring site exists and what services it must provide for the network. The researcher recommends that the monitoring of the network be automated to a point such that only one operator and his staff are required to 'control' the network. It is also felt that the position of 'Director of the LAN' be established as part of the CMS with authority over all aspects of network utilization. The tools recommended as part of the operator's workbench are seen as the basis upon which network monitoring, control, and management will be conducted. Without them, the accomplishment of the central monitoring site's mission will be questionable. To conclude, it is the researcher's opinion that the network operator's responsibilities we have identified in this Chapter, although not all encompassing, can be used as a basis upon which extended and more specific requirements can be built.

APPENDIX A

PROBLEM MANAGEMENT RECORD ENTRIES

Time and date of problem awareness

Type of equipment and serial number

Remarks about the nature of the problem

Logical name of the affected network element

Target date and time for problem resolution

Current problem status

Assesment of problems impact on network components

Cross reference to appropriate entry in configuration management database

Physical location of problem occurance and of elements reporting the problem

Date and time of problem resolution

Point of contact and phone number through which additional information concerning the problem can be obtained

APPENDIX B

CONFIGURATION MANAGEMENT RECORD ENTRIES

Item of equipment
Model and serial numbers
Physical address
Components logical name
Rental/Purchase price
Depreciation information
Installed/uninstalled status
Order number
Ship date
Lease start and end lease dates
Vendor name, phone number, and address
Associated node logical name
Item description/function
Remarks
Point of contact; name, location, telephone number
Building and room piece of equipment is in
Machine software was execiting on when problem was detected

LIST OF REFERENCES

1. Department of the Navy Fleet Material Support Office, System System Specification F94L0-001-9260-SS-SU01, *Stock Point Logistics Integrated Communications Environment(SPLICE)*, 2 February 1981.

2. Inman K.A., and Marthouse,R.C., *Local Area Computer Network Design Issues for Communications* M.S. Thesis, Naval Postgraduate School, Monterey, California, 1982.

3. Information Sciences Institute, University of Southern California, DOD Standard Transmission Protocol, Defense Advanced Research Projects Agency, January 1980.

4. Schneidewind, Norman F., "Functional Design of a Local Area Network for the Stock Point Logistics Integrated Communications Environment", *Naval Postgraduate School Document NPS-54-82-003* 30 September 1982.

5. Freeman, Richard B.,"Net Management Choices: Sidestream or Mainstream", *Data Communications*, August 1982.

6. Tobagi, Fouad A., and others, "Modeling and Measurement Techniques in Packet Communication Networks", *Proceeding of the IEEE*, V.66, N.11, November 1978.

7. Nutt,Gary J., "Tutorial: Computer System Monitors", *Computer*, November 1975.

8. Amer, Paul D., "A Measurement Center for the NBS Local Area Computer Network", *IEEE Transactions on Computers*, V.C-31, N.8, August 1982.

9. Tobagi, Fouad A., Lieberson, Stanley E., Kleinrock, L., "On Measurement Facilities In Packet Radio Systems", in National Computer Conference *AFIPS Conference Proceedings*, Vol.45, 1976.

10. Cole, Gerald D., "Performance Measurements on the ARPA Computer Network", *IEEE Transactions on Communications*, Vol. COM-20, No.3, June 1972.

11. Herman, James G., and Berstein, Susan L., *Monitoring, Control, and Management of the Defense Data Network*, Bolt, Beranek and Newman Inc., internal working paper, 1982.

12. Kleinrock, Leonard, and Naylor, William E., "On Measured Behavior of the ARPA Network", in National Computer Conference *AFIPS Conference Proceedings*, Vol. 43, 1976.

13. Department of the Navy Solicitation Document N66032-82-R-0007, *Acquisition of Hardware, Software and Services to Support the Stockpoint Logistics Integrated Communications Environment (SPLICE) Project at 62 Navy Stock Point Sites*, 1 March 1982.

14. Shoch, John F., and Hupp, Jon A., "Measured Performance of an Ethernet Local Network", *Communications of the ACM*, Vol. 23, No. 12, December 1980.

15. Sylor, Mark, *Network Performance Monitoring*, paper presented at Lawrence Livermore Laboratory, Livermore, CA, 19 May 1982.

16. Lynch, Tom, "DECnet Performance Measurement and Tuning", *Lawrence Livermore National Laboratory Local Users Group Newsletter*, July 1982.

17. Shoch, John F., Dalal, Yogen K., and Redell, David D., "Evolution of the Ethernet Local Computer Network", *Computer*, v. 15, n. 3, August 1982.

18. Metcalfe, Robert M., and Boggs, David R., "Ethernet: Distributed Packet Switching for Local Computer Networks", *Communications of the ACM*, v. 19, n. 7, July 1976.

19. Heiden, Hiedi B., and Duffield, Howard C., *Defense Data Network*, Defense Communications Agency internal working paper, 1982.

20. Ahlstrom, Tim, "Net Management Potential for Savings Dramatic", *Computerworld*, 27 September 1982.

21. Brice, Richard and Alexander, William, *A Network Performance Analyst's Workbench*, paper presented at Computer Network Performance Symposium, University of Maryland, 13 April 1982.

22. Gerla, M., and Kleinrock, L., *Closed Loop Stability Controls for S-ALOHA Satellite Communications*, presented at the Data Communications Symposium, Snowbird, Utah, September 1977.

23. McKenzie, Alexander A., *The ARPA Network Control Center*, paper presented at the 4th Data Communications Symposium, October 1975.

24. University of California Lawrence Livermore Laboratory Letter (Joe Requa): to Captain Craig E. Opel, Naval Postgraduate School, Subject: *Monitoring and Control of the Lawrence Livermore Laboratory Octopus Network*, 15 October 1982.

25. Codex Corporation, *Codex Distributed Network Control Systems 200 and 300 Planning Guide*, February 1982.

26. University of Delaware, Department of Computer and Information Sciences Letter (Dr. Paul Amer): to Captain Craig E. Opel, Naval Postgraduate School, Subject: *LAN Measurement and Performance Analysis*, 4 November 1982.

27. Gray, James P., "Network Services in Systems Network Architecture", *IEEE Transactions on Communications*, v. COM-25, n. 1, January 1977.

28. Cotton, Ira W., " Computer Network Interconnection", Proceedings of the Second Berkeley Workshop on Distributed Data Management, *Computer Networks*, May 1977.

29. Cerf, Vinton G., and Kirstein, Peter T., "Issues in Packet-Network Interconnection", *Proceedings of the IEEE*, V. 66, N. 11, November 1978.

30. Sheltzer, Alan, Hinden, Robert, and Bresica, Mike, "Connecting Different types of Networks with Gateways", *Data Communications*, August 1982.

31. Defense Communications Agency, *Defense Data Network Program Plan*, January 1982.

32. Poulos, John C., and Beavers, Alex N., *Network Operating Systems: A Concept and a Protocol*, internal working paper Booz, Allen and Hamilton Inc., Bethesda Maryland, 1980.

33. Leach, J.R., "Central Management Site for Users to Voice Problems Vital Step in Guiding Product Design", *Computerworld*, 27 September 1982.

34. Requa, Joseph E., *Octopus Network Monitoring and Measuring Project Preliminary Design*, system working paper Lawrence Livermore Laboratory, 7 October 1982.

35. Klingner, Thomas C., *Proposed FOCUS Scheduling Philosophy and Algorithm*, internal working paper Los Alamos National Laboratory, 17 June 1981.

36. Amer, Paul D., Rosenthal, Robert, and Toense, Robert, *NBS/ICST Measurement of a Local Area Computer Network*, National Bureau of Standards, Institute for Computer Sciences and Technology, paper submitted for publication in *Data Communications*, 4 November 1982.

37. Murphy, Jenanne L., "Centralized Control and Monitoring of a Distributed Local Computer Network", *Proceedings of the 6th Conference on Local Computer Networks*, 12 October, 1981.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3.	Professor Norman F. Schneidewind, Code 54Ss Department of Computer Sciences Naval Postgraduate School Monterey California 93940	2
4.	Lieutenant Colonel J. Mullane Marine Corps Representative Code 0309 Naval Postgraduate School Monterey, California 93940	1
5.	Professor Norm Lyons, Code 54Lb Department of Administrative Sciences Naval Postgraduate School Monterey, California 93940	1
6.	Captain Craig E. Opel 12589 Yardarm Place Woodbridge, Virginia 22192	2
7.	Dr. Sten Andler IBM Research K52/282 5600 Cottle Road San Jose, California 95139	1
8.	Mr. Joe Requa, L-60 University of California Lawrence Livermore Laboratory P.O. Box 808 Livermore, California 94550	1
9.	Mr. David Tolwinski CODEX Corporation 20 Cabot Boulevard Mansfield, Massachusetts 02048	1
10.	Dr. Paul Amer Department of Computer Science University of Delaware Newark, Delaware 19711	1
11.	Mr. Vic Russell Defense Communications Agency DDN/PMO B615 Washington D.C. 20305	1
12.	Cdr. Charpantidis Cosmas SMC #2291 Naval Postgraduate School Monterey, California 93940	1

13. Lcdr. Ted Case 1
Fleet Material Support Office
Code 94L
Mechanicsburg, Pennsylvania 17055

14. Lcdr. Dana Fuller 1
Commander, Naval Supply Systems Command
Code 0415A
Washington, D.C 20379

15. Ms. Mary Willoughby 1
P.O. Box 94
Mendocino, California 95460

200121

Thesis
05833 Opel
c.1
Network management
of the SPLICE computer
network.

MAY 13 85 30188
12 FEB 87 33381
19 NOV 87 32668

200121

Thesis
05833 Opel
c.1
Network management
of the SPLICE computer
network.

thes05833
Network management of the SPLICE compute
3 2768 001 97362 1
DUDLEY KNOX LIBRARY